Sparse Coding Techniques for Adjustable Lossy Compression of Baseband Signals in Centralized and Virtualized Communication Systems

Thibault Piana, Abdeldjalil Aissa El Bey, Matthieu Arzel, Alain Thomas

Sparse Coding Techniques for Adjustable Lossy Compression of Baseband Signals in Centralized and Virtualized Communication Systems

1st Thibault PIANA
Lab-STICC &
Space and Communication Lab

2nd Abdeldjalil Aïssa-El-Bey
IMT Atlantique,
Lab-STICC,
UMR CNRS 6285,
F-29238, Brest, France

3rd Matthieu Arzel
IMT Atlantique,
Lab-STICC,
UMR CNRS 6285,
F-29238, Brest, France

4th Alain Thomas
Safran Data Systems,
Space and Communication Lab,
Les Ulis, France

firstname.lastname@imt-atlantique.fr

***Abstract*—This paper presents an efficient lossy compression method based on sparse coding for received noisy signals (RX) in satellite communication systems. The emergence of ground station as a service (GSaaS) and centralized radio access networks (C-RAN) requires efficient data transfers between ground stations and datacenters. Our approach employs sparse coding to compress baseband signals via offline trained dictionaries. We analyze the loss introduced by compression and its impact on common modulation schemes across a Gaussian channel using the Bit Error Rate (BER) measurement. Additionally, the proposed method is compared to lossless linear prediction coding (LPC) and uniform quantization. Our study highlights the potential of lossy sparse coding to reduce bandwidth requirements while preserving signal integrity by offering a potential compression ratio of up to 8% higher than existing methods for the same degradation.**

***Index Terms*—dictionary learning, sparse coding, lossy compression, entropy coding, C-RAN, GSaaS**

I. Introduction

A. Context and Motivations

THE advent of Ground Station as a Service (GSaaS), has revolutionized the provision of services, particularly in the realm of Earth Observation. This innovation has democratized access to what was once prohibitively expensive infrastructure for many organizations, enabling them to leverage ground stations and extensive terrestrial network systems without the high investment. Drawing inspiration from the cost effectiveness and operational agility seen in mobile networks, these services have embraced the principles of centralized radio access networks (C-RAN). Centralization, a hallmark of C-RAN, offers notable advantages that include economical maintenance, optimized resource distribution, and improved efficiency in a widely deployed network. This approach necessitates the bulk of digital processing to be centralized within data centers, relegating base or ground stations to the role of merely facilitating the transmission and reception of raw signal samples to and from their analog front-ends. As depicted in Figure 1, this architecture underscores the dependency on complex raw sample transfers that encapsulate the baseband signal, demanding substantial bandwidth for network links that connect ground stations with data centers to handle both transmission (TX) and reception (RX) paths. The TX path is characterized by its noiseless transmission of signals to the antenna, whereas the RX path involves the reception of noisy signals from the antenna.

Parts of these research have been funded by *Safran Data Systems*.

To mitigate bandwidth demand and reduce consumption, employing compression strategies is considered efficient. Although several lossless compression methods have been explored [1]–[3], the exploration of lossy compression, which is notably complex, has been predominantly concentrated on traditional techniques such as quantization and resampling [4]. This article focuses on a new approach based on sparse coding techniques.

B. Introduction to Sparse Coding

Sparse Coding is a powerful signal processing technique that aims to represent signals as a linear combination of a small number of elementary building blocks, also known as atoms, from an dictionary. Mathematically, given a signal $\boldsymbol{y} \in \mathbb{R}^M$ and a dictionary $\boldsymbol{D} \in \mathbb{R}^{M \times K}$, where $K \geq M$, the goal is to find a sparse representation $\boldsymbol{x} \in \mathbb{R}^K$ such that $\boldsymbol{y} \approx \boldsymbol{D}\boldsymbol{x}$. The sparsity constraint implies that only a few entries in $\boldsymbol{x}$ are nonzero, making it possible to efficiently represent and transmit the signal using a reduced number of parameters.

In the context of satellite communication systems, sparse coding can be used to find an efficient representation of the received signals using their specific spectral and temporal characteristics. By representing the incoming signals as a sparse combination of atoms in a dictionary $\boldsymbol{D}$, we can efficiently compress the data transmitted between the antenna and the data center.

C. Contribution

In this paper, we focus on the compression method suitable for the receiver side (RX) of a satellite transceiver, where signals are subject to noise. Our aim is to present an efficient, lossy compression method based on sparse coding to reduce the required bandwidth while preserving the integrity of the

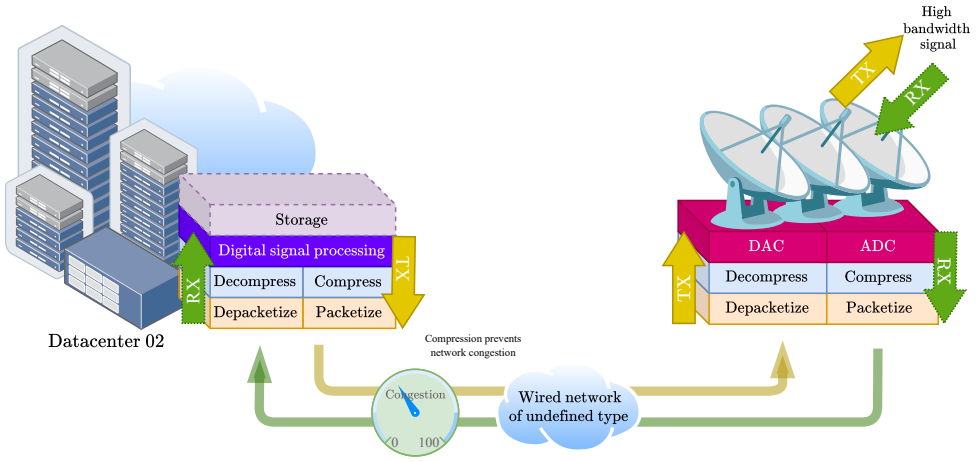

Fig. 1. Overview of a remote/centralized modem processing architecture used in satellite communications systems. The figure shows a comparison between the traditional (top) C-RAN architecture and the enhanced (bottom) version with compression. The compressed C-RAN architecture results in reduced network congestion and lower overall data rates, while also providing increased capacity that allows multiplexing of input data over the same network. These improvements have the potential to enhance network efficiency and performance.

signal. We also provide a comprehensive analysis of the loss introduced by this compression, examining its impact on various types of modulation across a Gaussian channel using metrics from digital communications (bit error rate, E_s/N_0...). Finally, we present theoretical insights into the expected compression performance under these conditions and compare them with other well-used methods such as lossless linear prediction coding (LPC) [1], [2], [5] or simple quantization.

II. Lossy Compression With Sparse Coding

We define the compression ratio of an algorithm applied to a dataset as follows:

$$C_{\text{ratio}} = 1 - \frac{\text{Compressed size (in bits)}}{\text{Original size (in bits)}}$$

Furthermore, we characterize the impact of a compression algorithm as the displacement, measured in decibels (dB), towards the right of the signal's Bit Error Rate (BER) curve (Figure 3).

A. Dictionary Learning

In dictionary learning, we aim to find a dictionary $\boldsymbol{D}_{\tau_0} \in \mathbb{R}^{M\times K}$ and a sparse representation $\boldsymbol{x} \in \mathbb{R}^K$ to approximate input signal $\boldsymbol{y} \in \mathbb{R}^M$. We solve the following optimization problem:

$$\begin{aligned} \underset{\boldsymbol{D}_{\tau_0},\boldsymbol{x}}{\arg\min} \quad & \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{D}_{\tau_0}\boldsymbol{x}\|_2^2 \\ \text{subject to} \quad & \|\boldsymbol{x}\|_0 \leq \tau_0, \\ & \|\boldsymbol{d}_i\|_2 = 1, ; i = 1, \ldots, K, \end{aligned} \tag{1}$$

Where $\|\cdot\|_2$ is the Euclidean norm, $\|\cdot\|_0$ is the pseudo-norm counting nonzero elements in a vector, $\boldsymbol{d}_i$ is the i-th column of $\boldsymbol{D}_{\tau_0}$, and $\tau_0 > 0$ represents a sparsity constraint parameter. When $K = M$, $\boldsymbol{D}_{\tau_0}$ is said to be *complete* and when $K > M$, $\boldsymbol{D}_{\tau_0}$ is said to be *overcomplete*.

Familiar techniques such as K-SVD [6] or Online Dictionary Learning (ODL) [7] are applicable to learn the dictionary $\boldsymbol{D}$. The choice of method, along with the amount of data at hand, determines whether the dictionary learning (DL) process is carried out *online* (direct training on $\boldsymbol{y}$) or *offline* (using a dataset that simulates the properties of $\boldsymbol{y}$).

Regarding the optimization problem to find $\boldsymbol{x}$, there are numerous sparse coding algorithms, the most known being Orthogonal Matching Pursuit (OMP) [8]. Other methods such as LASSO [9] or Coordinate Descent (CD) [7] use ℓ_1 regularization [10], [11] to achieve the same goal at lower complexity.

B. Managing loss when using sparse coding

Since only a few entries in the sparse representation vector are nonzero, the signal can be transmitted or stored using a reduced number of samples, resulting in efficient compression. However, the benefits of sparse representations come with a trade-off: The resulting compression is lossy in nature, which implies that some degree of information loss may occur during the compression/decompression process.

In contrast to image processing, where this compression scheme is already well known and used [12]–[16], digital communications require unbiased and meticulous considerations of this loss, which can then be considered as *compression noise*. Factors such as the noise statistical distribution law and power spectral density (PSD), whether it is white or colored noise, play crucial roles in evaluating the impact of noise on the overall system performance.

Furthermore, the intricate nature of receivers and the sequence of nonlinear stages, such as synchronization, interleaving, and error correction codes, contribute to the highly nonlinear decoding process. As a result, traditional metrics such as the mean squared error (MSE) or the mean absolute error (MAE) cannot be relied upon for analysis. Instead, it is necessary to undergo the complete decoding process to accurately measure the Bit Error Rate (BER).

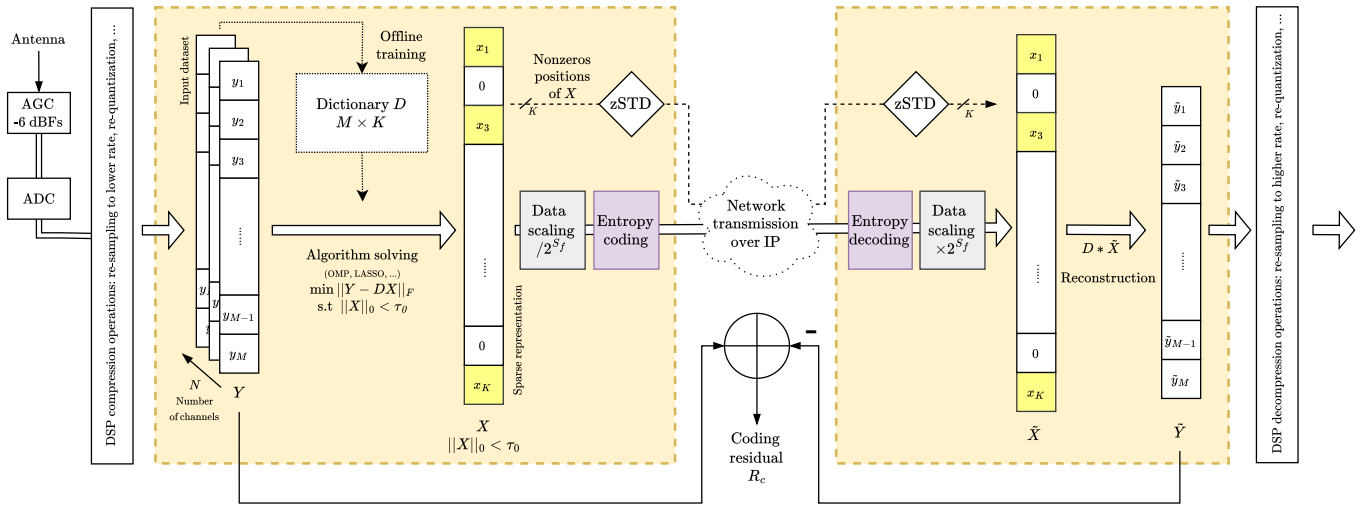

Fig. 2. Proposed compression architecture diagram. The compression ratio is determined by the scaling factor S_f and the number of nonzero values τ_0.

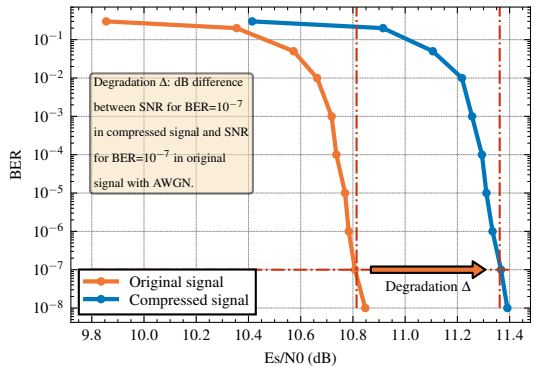

Fig. 3. Measure of the degradation (in dB). To ensure convergence, we use both FEC and interleaver.

To optimize the trade-off between compression efficiency, computation time and reconstruction quality, many parameters can be fine-tuned, such as the number of samples M, the dictionary overcompleteness K, the number of nonzero coefficients τ_0 or the choice of the sparse coding algorithm. Our research presents results and ways to determine the best parameters depending on requirements.

C. The art of sending sparse data

We examine two main approaches to enhance the efficiency of data compression when dealing with sparse data.

The conventional approach, historically used in image processing, involves the transmission of data in its complete form, incorporating both nonzero and zero values. This method fundamentally relies on entropy coding, like Huffman, using the statistical distribution properties of sparse data to aid in compression. The effectiveness of this strategy depends primarily on the capacity of Huffman coding to construct an optimally condensed tree, which effectively encapsulates the distribution between nonzero and zero elements.

In contrast, this research introduces an optimized approach, deviating from conventional approaches. This innovative strategy is based on selective transmission, where only nonzero samples and their corresponding positions are transmitted. The nonzero data elements are further processed through Huffman coding, optimizing the efficiency of transmission by exploiting their statistical distribution. Simultaneously, the positions of these nonzero elements are compressed using a classical dictionary compressor, significantly reducing the overhead associated with their transmission (see below). By avoiding the transfer of zero values and utilizing these two compression methods simultaneously, we noticed that this method inherently reduces the volume of data that needs to be transmitted.

D. The optimized way to send nonzero values positions

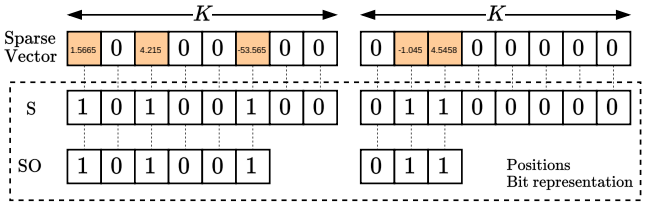

Fig. 4. Two ways to send nonzeros positions. Each row represents the actual bits transmitted to transmit nonzero values position, for each method.

Figure 4 presents the different way we can use to send nonzero positions to the decompressor to reconstruct the data:

1) **The S form** is the simplest and non-optimized method for transmitting positions. It involves sending a vector of K bits, where a 0 represents a value of 0 and a 1 represents a nonzero value.
2) **SO form** is an optimized form in which we use the fact that there are precisely τ_0 nonzero values. We then stop the bit vector at the τ_0th bit equal to 1 and then start directly the next vector.
3) **SOC or SC forms** are similar to the previous forms, but the resulting bit vector is passed through a lossless

dictionary compressor: zSTD [17]. Figure 5 shows how the application of this method actually decreases the number of bits needed to transmit nonzero positions.

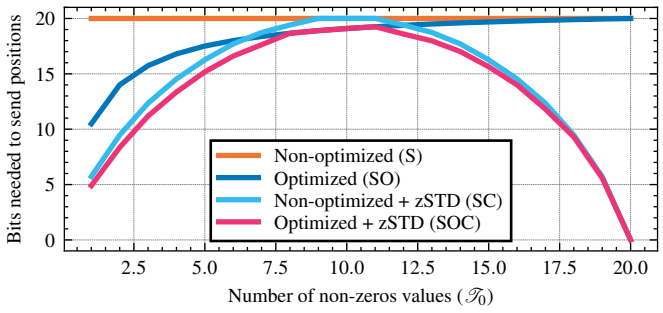

Fig. 5. Evolution of the number of bits to send for each channel in order to transmit the position of nonzeros values to the decompressor ($M = K = 20$). zSTD could actually be replaced by another compression mechanism (gZIP, LZMA,...).

E. Application to digital communications

In this summary, we examine an 8-PSK baseband signal shaped using a 0.25 roll-off root-raised-cosine (RRC) filter, a standard waveform used in space telecommunications. After sampling the signal with the analog-to-digital converter (ADC), we employ the compression chain depicted in Figure 2. Table I lists all the main parameters chosen for the simulation.

TABLE I
SIMULATION PARAMETERS

AGC target	-6 dBFs
Interleaver size	324,000
FEC	LDPC 5/6 (DVB-S2 mod15)
BER goal	10^{-6}
Quantization	8 bits
Oversampling	2 samples per symbol
M	120
K	120
N	44,000

The parameters are based on typical settings and constraints for DVB-S2 signals. The AGC target of -6 dBFs accommodates the typical noise range for DVB-S2. A 324,000-size interleaver covers five DVB-S2 frames, providing a safety margin for uncertainty in data segmentation for sparse coding. The 5/6 LDPC FEC under modcod 15 is typical for moderate SNR regions. An 8-bit quantization is standard, and 2 samples per symbol for oversampling align with common practice. M is set at 120, balancing the complexity of the algorithm and the effectiveness of compression against degradation, particularly since the efficient compression zones for τ_0 are narrow (based on Figures 6 and 7, it can be observed that the effective range of values for τ_0 is approximately between 20 and 60). K equals M since the usage of overcomplete dictionaries did not show any improvement in the final results, simplifying our approach.

The dictionary for each value of τ_0 is learned *offline* before simulation on a noiseless signal of the same kind according to Equation 1 using Mairal's ODL algorithm [7]. The actual implementation of the OMP used is the Order Magnitude Matching Pursuit (ORMP) algorithm, which has slightly better results than the classical OMP implementation. Here is the sequence of procedures employed to compress the IQ data:

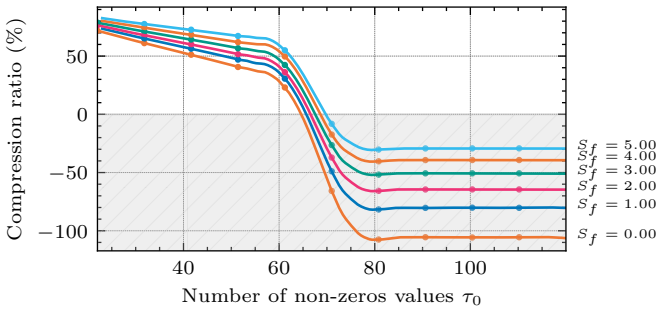

Fig. 6. Evolution of compression ratio depending on each parameters (for SOC). $M = K = 120$. For $\tau_0 \gtrsim 60$, there is no compression gain.

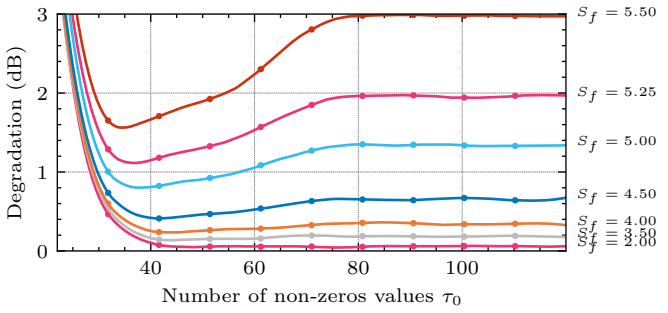

Fig. 7. Evolution of degradation depending on each parameters (for SOC). $M = K = 120$. When $\tau_0 \lesssim 30$, the degradation begins to grow exponentially.

Step 1: Based on the desired level of degradation and the target compression ratio, we select a pair of parameters (τ_0, S_f). Given the infinite number of pairs for a particular degradation, it is necessary to identify the pair that, for a fixed degradation, maximizes the compression ratio. This involves examining all potential pairs and choosing the most optimal one. The preliminary investigation step, illustrated in Figure 8, is carried out offline before compression occurs.

Step 2: The samples are divided into groups of M adjacent samples, and a sparse coding operation is performed on them using a dictionary trained offline $\boldsymbol{D}_{\boldsymbol{\tau_0}}$.

Step 3: Zeros are removed from the sparse representation, and nonzero values pass through an entropy coder (Huffman coder).

Step 4: In order to reconstruct the sparse vector, nonzero value positions are also sent over the network, but are first packed and compressed using a dictionary algorithm according to Section II-D. We chose zSTD

(zStandard) [17] because of its trade-off between speed and performance.

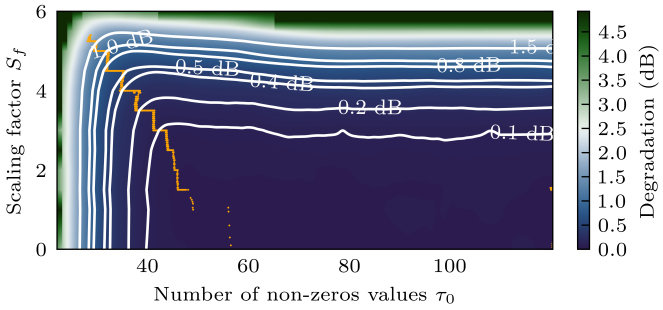

Fig. 8. Evolution of degradation depending on each parameters. Orange dots represent the optimal parameter combinations that maximize compression.

Once reconstructed, the data are passed through the entire demodulation process, including the interleaver and forward error correction (FEC) processes. These two processes are actually the most important part of the presented method. The interleaver will spread the compression error across the entire signal, while the FEC will ensure that the BER will reach the target value even when strong degradation is applied.

F. Results

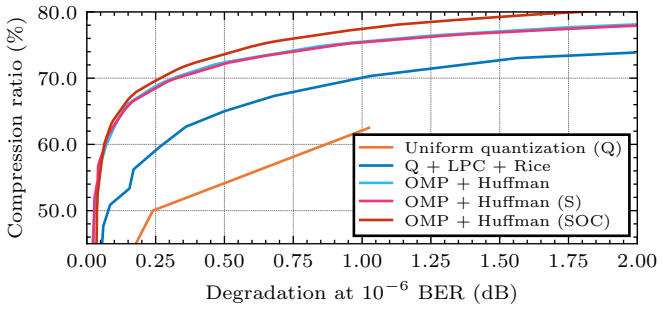

Fig. 9. Expected compression ratio for each degradation. Simulations were performed for an 8-PSK signal with a 5/6 LDPC FEC. See Table I for the list of all simulations parameters.

To perform our simulations, we generate an artificial signal with Matlab, add AWGN noise for a specific E_s/N_0 SNR, compress and decompress it using our architecture, and finally measure the output bit error rate (BER). Figure 9 shows that the proposed architecture can offer up to 8% higher compression ratio than classical methods for equivalent degradation. It also provides more flexibility than pure quantization, offering a more reachable compression ratio range depending on M.

III. Conclusion

The proposed data compression method shows better results compared to previous techniques [1], [4], achieving remarkable compression ratios and a highly controllable loss. Future studies could investigate alternative compressor architectures or different entropy encoders, such as Golomb-Rice coding [18], [19], to further enhance the efficiency and versatility of the method while offering improvements in the practical deployment capabilities of the solution, including hardware and software implementations.

References

[1] L. Ramalho, M. N. Fonseca, A. Klautau, C. Lu, M. Berg, E. Trojer, and S. Host, "An LPC-Based Fronthaul Compression Scheme," *IEEE Communications Letters*, vol. 21, pp. 318–321, Feb. 2017.

[2] L. Ramalho, I. Freire, C. Lu, M. Berg, and A. Klautau, "Improved LPC-Based Fronthaul Compression With High Rate Adaptation Resolution," *IEEE Communications Letters*, vol. 22, pp. 458–461, Mar. 2018.

[3] T. Piana, M. Arzel, A. Aissa El Bey, and A. Thomas, "Neural Networks as Catalysts for Enhanced Lossless Compression in Satellite Communications," in *International Conference on Emerging Technologies for Communication, ICETC 2023*, p. 4, ICETC, Nov. 2023.

[4] A. Shehata, M. Crussiere, and P. Mary, "Analysis of Baseband IQ Data Compression Methods for Centralized RAN," in *2020 28th European Signal Processing Conference (EUSIPCO)*, (Amsterdam, Netherlands), pp. 1762–1766, IEEE, Jan. 2021.

[5] L. Ramalho, J. Fortuna, C. Lu, M. Berg, I. Almeida, and A. Klautau, "Rate Control for Entropy-Coded LPC: Application to Packet-Based Fronthauling," *IEEE Transactions on Communications*, pp. 1–1, 2022.

[6] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, p. 12, 2006.

[7] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning*, (Montreal Quebec Canada), pp. 689–696, ACM, June 2009.

[8] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, pp. 40–44 vol.1, Nov. 1993.

[9] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996.

[10] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal l1-norm solution is also the sparsest solution," *Communications on Pure and Applied Mathematics*, vol. 59, pp. 797–829, June 2006.

[11] D. L. Donoho, "For most large underdetermined systems of equations, the minimal l1-norm near-solution approximates the sparsest near-solution," *Communications on Pure and Applied Mathematics*, vol. 59, pp. 907–934, July 2006.

[12] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. New York, NY: Springer New York, 2010.

[13] H. Wang and T. Celik, "Sparse Representation-Based Hyperspectral Data Processing: Lossy Compression," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, pp. 2036–2045, May 2017.

[14] X. Zhang, W. Lin, Y. Zhang, S. Wang, S. Ma, L. Duan, and W. Gao, "Rate-Distortion Optimized Sparse Coding With Ordered Dictionary for Image Set Compression," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, pp. 3387–3397, Dec. 2018.

[15] T. Kavitha and K. Jaya Sankar, "A Fast and Efficient Lossless Compression Technique for Greyscale Images," *International Journal of Circuits, Systems and Signal Processing*, vol. 14, Apr. 2020.

[16] M. Kalluri, M. Jiang, N. Ling, J. Zheng, and P. Zhang, "Adaptive RD Optimal Sparse Coding With Quantization for Image Compression," *IEEE Transactions on Multimedia*, vol. 21, pp. 39–50, Jan. 2019.

[17] Y. Collet and M. Kucherawy, "Zstandard Compression and the application/zstd Media Type," Request for Comments RFC 8478, Internet Engineering Task Force, Oct. 2018.

[18] R. Rice and J. Plaunt, "Adaptive Variable-Length Coding for Efficient Compression of Spacecraft Television Data," *IEEE Transactions on Communication Technology*, vol. 19, pp. 889–897, Dec. 1971.

[19] B. McKenzie and T. Bell, "Compression of sparse matrices by blocked Rice coding," *IEEE Transactions on Information Theory*, vol. 47, pp. 1223–1230, Mar. 2001.