Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/35/20230385

A comparative analysis of convolutional coding and block coding techniques over satellite communication

Yuchen Hou

College of Information Engineering, Nanjing University of Aeronautics and Astronautics, 29 General Avenue, Nanjing, 211106, China

042100824@nuaa.edu.cn

Abstract. Reliable data transmission is crucial in the field of satellite communication, and many coding schemes are used to improve this reliability. Two popular coding techniques: convolutional codes and block codes were investigated and compared in this paper. Utilizing a simulated environment, the author investigates the bit error rate (BER) performance of both coding strategies along with a baseline case of no coding. The analysis reveals distinct characteristics and performance attributes of the two coding techniques across a range of signal-to-noise rate (SNR) values. Convolutional codes, with their inherent memory, offer error protection across sequences, while block codes, particularly a Hamming-like 7,3 implementation, provide localized protection for individual blocks. The findings of this study contribute to the understanding of error control strategies in satellite communication, guiding the selection and design of coding methods suitable for various operational scenarios and requirements.

Keywords: convolutional coding, block coding, satellite communication.

1. Introduction

Satellite communication demands highly reliable transmission due to vast distances and potential interferences. While increasing the transmission power is a straightforward solution, it's not always feasible for satellite systems due to limited onboard power resources. Consequently, the focus shifts to error-correction techniques, with coding schemes emerging as the most potent strategy.

The bit error rate (BER) serves as a critical metric for these systems, with a lower BER denoting higher reliability. Among the various coding methods, convolutional and block coding have garnered significant attention. Convolutional codes are adept at handling burst errors, while block codes, like the Hamming code, are tailored for random bit errors [1].

In this paper, the author compares these two techniques against a no-coding scenario in the context of satellite communication, seeking to elucidate the optimal coding strategy to ensure efficient and reliable transmissions.

2. Concepts of convolutional codes and block codes

Convolutional codes are a type of error-correcting code where each input bit affects multiple output bits, utilizing memory elements to encode sequences over time [2]. The encoded output depends not just on the current input, but also on previous inputs, making it suitable for continuous data streams.

DOI: 10.54254/2755-2721/35/20230385

Block codes, on the other hand, segment data into fixed-size blocks and add redundancy bits to each block. The redundancy allows for both error detection and, in many cases, error correction. Examples include Hamming and Reed-Solomon codes [3]. They're best suited for situations with fixed-size data packets.

3. Transmission characteristics in satellite links

3.1. Overview of satellite links

Satellite links provide a crucial communication bridge between distant terrestrial stations by utilizing a satellite as a relay. Operating primarily in microwave frequency bands, these links enable global coverage, overcoming the limitations of terrestrial networks. The very nature of satellite communication—transmitting signals through the Earth's atmosphere and vast expanses of space—introduces challenges like signal attenuation, delay, and interference. Nevertheless, advancements in satellite technology and adaptive modulation techniques have enhanced the reliability and speed of these links, making them indispensable in modern communication, spanning from television broadcasting to global internet connectivity.

3.2. Atmospheric attenuation and multipath effects

3.2.1. Introduction to atmospheric challenges. Satellite communication signals, while traversing through the Earth's atmosphere, are susceptible to various forms of attenuation and distortions.

3.2.2. Atmospheric attenuation. One of the primary culprits is atmospheric attenuation, which arises due to the absorption and scattering of signals by atmospheric gases, rain, and clouds. Particularly in the higher frequency bands, like the Ku and Ka, rain-induced attenuation can be especially pronounced, causing significant degradation in link quality during adverse weather conditions.

3.2.3. Causes of multipath effects. Multipath effects present another challenge [4]. As the name suggests, signals can take multiple paths between the transmitter and receiver, typically due to the reflection or scattering of terrestrial objects or atmospheric anomalies. When these varied signals converge at the receiver, they can interfere constructively or destructively, leading to signal fading or enhancement.

3.2.4. Multipath in satellite links. In satellite links, ionospheric reflections can also contribute to multipath effects, especially at lower frequencies.

3.3. Noise and interference

In satellite communication, noise and interference are two primary adversaries that can degrade signal quality, leading to errors during data reception. Noise, an unwanted random signal, predominantly arises from thermal sources in the satellite's receiver and the Earth's atmosphere. It can be characterized by a parameter called Noise Temperature, which reflects the total thermal noise present in the system.

Interference, on the other hand, is an unwanted deterministic signal that overlaps with the desired satellite signal. It can originate from various sources: terrestrial transmitters broadcasting in the same or adjacent frequencies (co-channel interference), neighbouring satellites in the same orbital slots (adjacent satellite interference), or even the same satellite when a signal reflects off the Earth and travels back (echo interference).

4. Simulation platform development using MATLAB

The flow chart of the whole program is shown in Figures 1 and 2.

DOI: 10.54254/2755-2721/35/20230385

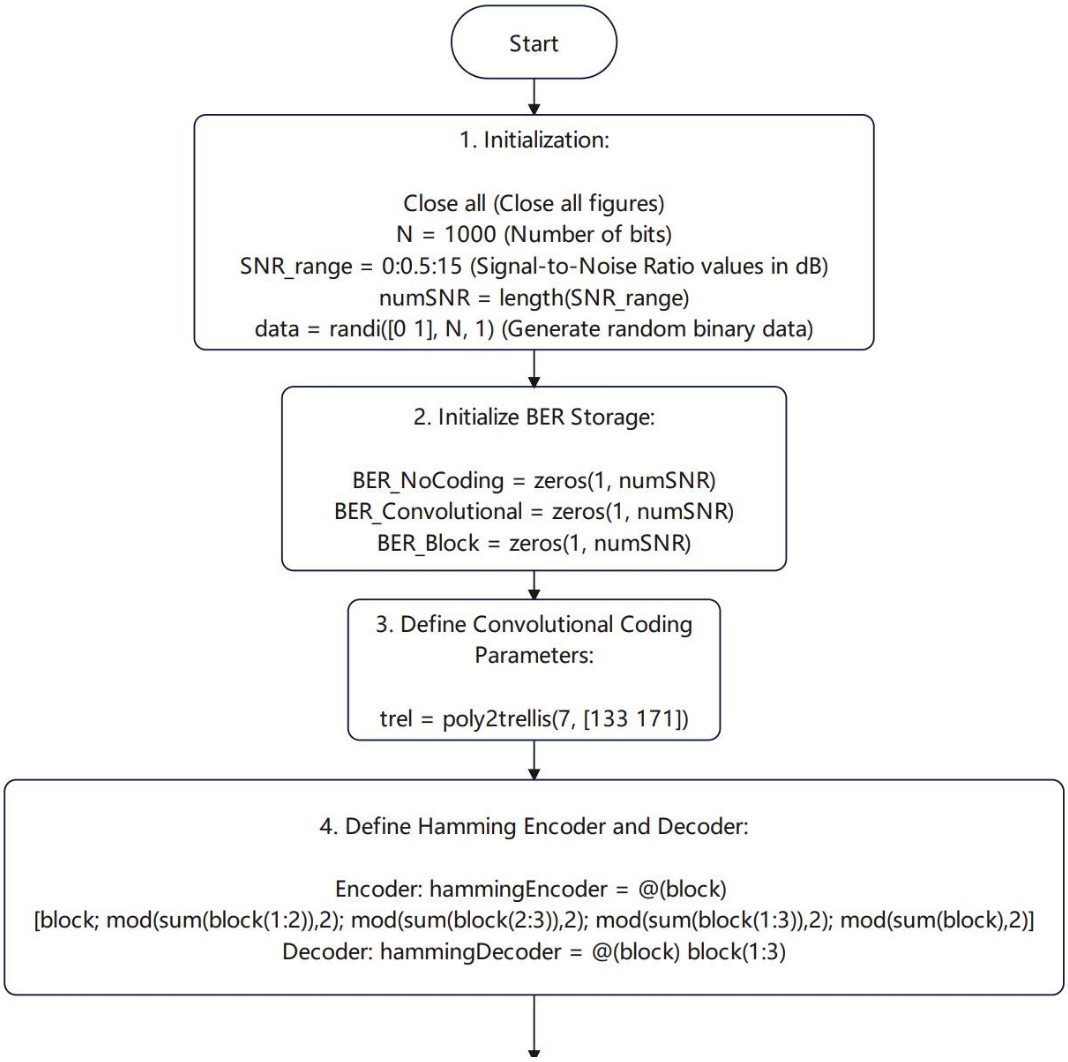

Figure 1. Flow chart (part 1).

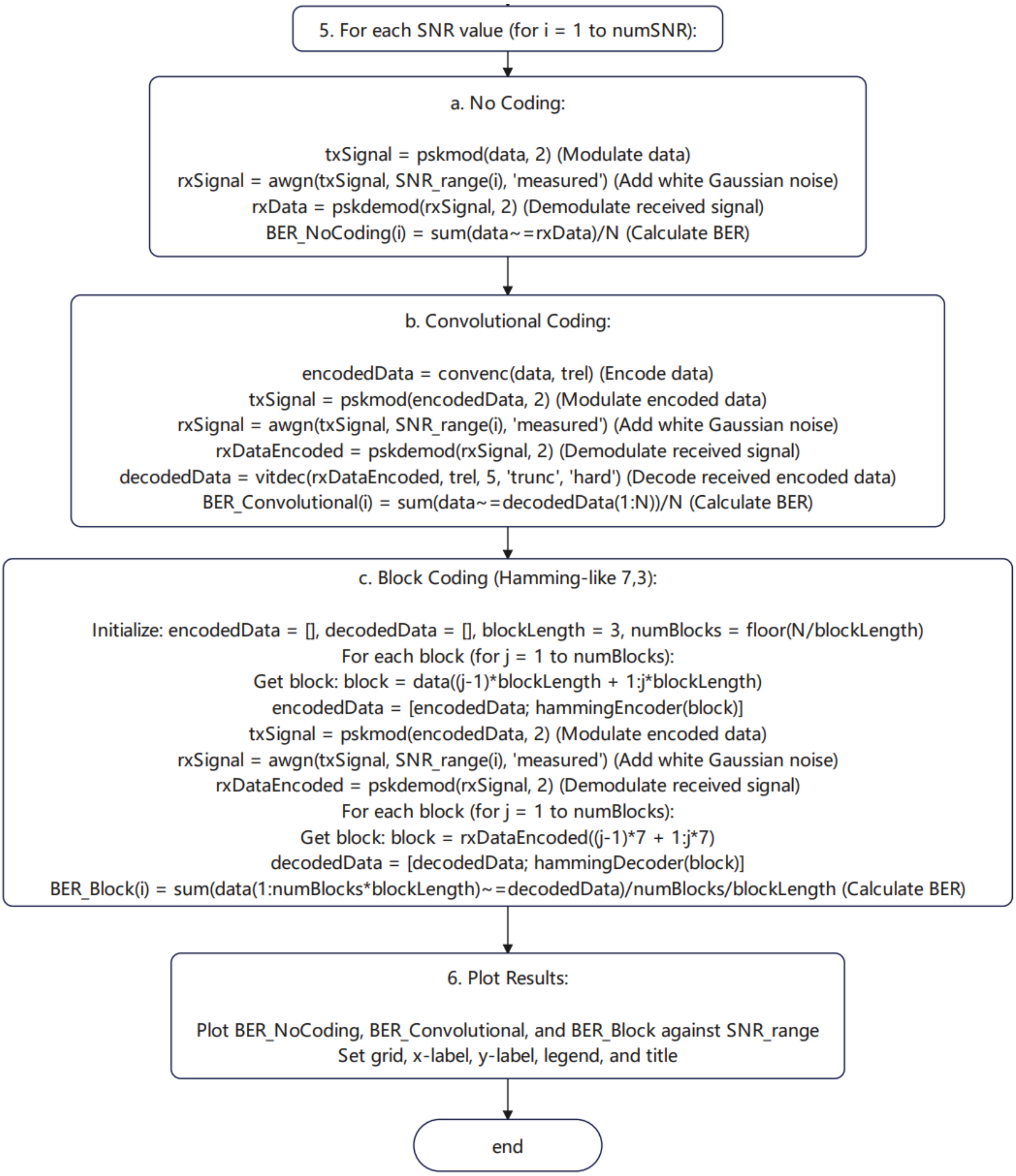

Figure 2. Flow chart (part 2).

4.1. *Platform design process*

4.1.1. Parameter initialization. In the simulation developed by the author, initializing the correct parameters is crucial for accurately modelling and evaluating the coding techniques in satellite communication. The initialization phase includes three key components.

Data Bit Length (N):

The author defines the variable N to represent the number of bits in the data sequence to be transmitted. For the given simulation, N is set to 1000, meaning that the input data sequence consists of 1000 randomly generated binary bits. This parameter determines the size of the data set that will be processed through the encoding, transmission, and decoding stages.

SNR Value Ranges:

Signal-to-noise rate (SNR) is a vital parameter in the simulation, indicating the relationship between the power of the signal and the power of the noise affecting the signal [5]. In the code, SNR range is defined from 0 to 15 dB, incremented by 0.5 dB. By varying the SNR, the author can evaluate how different levels of noise impact the BER for various coding techniques.

BER Storage Mechanism:

To record and analyse the BER for different coding techniques under various SNR conditions, the author employs storage vectors BER_No_Coding, BER_Convolutional, and BER_Block. Each of these vectors is initialized to store the BER corresponding to the SNR values defined earlier. The vectors are filled during the simulation's execution to hold the calculated BER for the no coding scenario, the convolutional coding, and the block coding, respectively.

By meticulously setting these parameters, the author has established a foundational framework for the simulation, ensuring that the subsequent encoding, modulation, and decoding stages can be executed accurately and meaningfully to reflect real-world satellite communication environments.

4.1.2. Encoder design & implementation. In the quest to evaluate the performance of various coding techniques for satellite communication, the author meticulously designed and implemented two distinct encoder strategies: the Convolutional encoder and a Simplified Hamming block encoder.

Convolutional Coding Principles and Parameters:

The principle of convolutional coding revolves around a state-machine based encoding approach, where each bit in the output is generated as a function of the current and previous input bits [6]. This provides a robust error-correction mechanism, as the resulting encoded sequence carries information from multiple original bits, offering redundancy.

For the design in MATLAB, the author used the built-in poly2trellis function to specify the convolutional encoder's trellis structure. The trellis was configured with a constraint length of 7 and generator polynomials of [133 171], which are common parameters for satellite communication due to their error-correcting capabilities. Specifically, the code is defined as:

```
trel = poly2trellis(7, [133 171]);
```

With this trellis structure in place, the data was then encoded using MATLAB's convenc function. This encapsulates the essence of convolutional coding by transforming the original data into a sequence that is resilient against errors during transmission.

Simplified Hamming Block Encoding Approach:

In contrast to the convolutional coding approach, block coding, specifically the Hamming scheme, segments the original data into distinct blocks and appends redundant bits to each block, facilitating error detection and correction [7].

The author formulated a simplified version of the Hamming code for a (7,3) configuration. This implies that for every block of 3 data bits, an additional 4 redundant bits are added, resulting in a 7-bit encoded block. The encoding approach was defined by:

```
hammingEncoder = @(block) [block; mod(sum(block(1:2)),2); mod(sum(block(2:3)),2); mod(sum(block(1:3)),2); mod(sum(block),2)];
```

The function takes a 3-bit block and calculates four parity bits based on combinations of the original bits. The entire original data sequence is then processed in 3-bit blocks, each of which is transformed using this Hamming encoder function.

The result of the block encoding approach is a sequence that's approximately 7/3 times longer than the original, but with the added redundancy providing the necessary framework for error detection and potential correction during the decoding phase.

DOI: 10.54254/2755-2721/35/20230385

In sum, through meticulous design and leveraging MATLAB's powerful functions, the author set the stage for a rigorous evaluation of the merits and drawbacks of both convolutional and block encoding strategies in satellite communication contexts.

4.1.3. Satellite channel modelling. In the process of designing a robust simulation platform, the author opted to model the satellite communication channel using two integral components: Phase Shift Keying (PSK) modulation and the Additive White Gaussian Noise (AWGN) model [8].

PSK Modulation:

The author employs the PSK modulation [9], a digital modulation scheme that conveys data by changing the phase of a reference signal (the carrier wave). This method is particularly effective for satellite communication, owing to its ability to retain the bandwidth efficiency while battling against the noise in the communication channel. In the given simulation, the data, represented as a binary sequence, undergoes binary PSK modulation. This is evident from the lines of code:

```
txSignal = pskmod(data, 2);
rxData = pskdemod(rxSignal, 2);
```

Here, 'pskmod' facilitates the modulation of the input data, whereas 'pskdemod' aids in demodulating the received signal back to its binary form.

AWGN Model:

Satellite channels are often plagued with various types of noise, with the AWGN being a predominant type. The AWGN model introduces random noise into the signal, simulating real-world transmission challenges. The author has incorporated the AWGN channel in the simulation to assess the performance of various coding techniques under noisy conditions. This is achieved using the MATLAB function 'awgn', as observed from:

```
rxSignal = awgn(txSignal, SNR_range(i), 'measured');
```

Here, the transmitted signal (txSignal) is subjected to AWGN based on the specific SNR under consideration. By iterating over a range of SNR values, the author systematically evaluates how the coding schemes fare against increasing noise levels.

In summary, the author's adoption of PSK modulation and the AWGN model provides a realistic depiction of a satellite communication channel, allowing for a comprehensive evaluation of the coding strategies under varying channel conditions.

4.1.4. Decoder implementation. The author implemented a comprehensive decoding system to ensure the accurate retrieval of original data. The decoding process is delineated into three distinct stages that cater to various encoding techniques. Here is a systematic breakdown:

PSK Demodulation:

For each SNR value in the considered range, the received signal undergoes a PSK demodulation. In the code, this is realized using the pskdemod function. This demodulation technique is essential for converting the received modulated signal back into its binary form. The PSK demodulation operates on the received signal, rxSignal, which has been exposed to AWGN in the satellite communication channel.

```
rxData = pskdemod(rxSignal, 2);
```

Viterbi Algorithm for Convolutional Code Decoding:

For data encoded using the convolutional code, a distinct decoding process is applied. The Viterbi algorithm, a dynamic programming algorithm known for its efficiency in decoding convolutional codes, is employed [10]. In the MATLAB environment, this decoding is executed using the vitdec function. The function requires the received data (rxDataEncoded), the trellis structure (trel), and other parameters like the traceback depth and decoding method. The convolutionally decoded data is then compared with the original data to compute the BER for this encoding scheme.

```
decodedData = vitdec(rxDataEncoded, trel, 5, 'trunc', 'hard');
```

Block Decoding for Hamming-like Scheme:

For the block coding approach, the Hamming-like 7,3 scheme is adopted. For each block of received encoded data, the implemented block decoder extracts the first three bits, which represent the original data. This simple block decoding is achieved using the hammingDecoder function. The author designed this function based on the Hamming-like 7,3 coding scheme, which entails extracting the primary data bits while omitting the error-checking bits [11].

```
decodedData = [];
for j = 1:numBlocks
    block = rxDataEncoded((j-1)*7 + 1:j*7);
    decodedData = [decodedData; hammingDecoder(block)];
end
```

In summation, the decoder implementation is intricately designed to cater to both convolutional and block coding techniques. By leveraging MATLAB's efficient functions and incorporating custom decoding methods, the author ensured the accurate recovery of transmitted data from varying encoding schemes and channel conditions.

4.2. Simulation result analysis methodology

4.2.1. Introduction to simulation analysis. Upon establishing the simulation platform in MATLAB, the author applied a meticulous approach to the analysis of the results. This section outlines the methodology adopted to comprehend the performance of the implemented coding techniques in satellite communication.

4.2.2. Importance of BER in communication systems. The BER signifies the ratio of erroneous bits received to the total number of bits transmitted. It is a quintessential metric to gauge the reliability and efficacy of a communication system, particularly in the presence of noise and interference.

4.2.3. Methodology for BER computation. Without any coding:

For each SNR value in the specified range, the author modulated the data using PSK, introduced noise via the AWGN model, and then demodulated the received signal. The number of erroneous bits was then determined by comparing the received data to the original transmitted data. The BER was computed by dividing the number of errors by the total bits, N.

For Convolutional Coding:

After encoding the data using a convolutional encoder, the author employed the same PSK modulation, introduced noise, and demodulated the received signal. The received encoded data was then decoded using the Viterbi algorithm. The BER was ascertained by comparing the decoded data to the original data over the length N.

For Block Coding (Hamming-like 7,3):

The data was partitioned into blocks and each block was encoded using a simplified Hamming-like scheme. The encoded data underwent PSK modulation, noise introduction, and subsequent demodulation. The received block-coded data was then decoded block by block. The BER for this scheme was calculated by contrasting the decoded data with the original data and normalizing it with the product of the number of blocks and block length.

4.2.4. SNR versus BER visualization. Visualization serves as a pivotal tool in comprehending the interplay between signal quality and error rates.

A logarithmic plot was set up using the semilogy function in MATLAB, which helps in discerning variations in BER, especially when it spans several orders of magnitude.

The SNR values in dB constituted the x-axis, while the computed BER values for each SNR formed the y-axis.

The author plotted three curves representing the BER for "No Coding," "Convolutional Coding," and "Block Coding (Hamming-like 7,3)." Each curve was adorned with distinct markers for clear differentiation.

DOI: 10.54254/2755-2721/35/20230385

Grid lines were added to the plot to facilitate a precise reading of values. The plot was also annotated with labels for the x-axis (SNR (dB)), y-axis , a legend detailing each curve, and an overarching title encapsulating the theme of the visualization.

5. Execution of Simulation & Performance Analysis

5.1. Design and configuration of simulation trials

5.1.1. Introduction. The author designed a comprehensive simulation using MATLAB to evaluate the impact of various coding schemes on BER in a noisy communication channel, particularly focusing on satellite communication.

5.1.2. Scenario without coding. Initially, the author set up a baseline scenario without any coding mechanism. A total of N = 1000 random binary bits were generated using MATLAB's randi function. These bits were then modulated using PSK with the help of the pskmod function. To simulate the effects of noise prevalent in a real-world satellite channel, the author introduced AWGN to the transmitted signal using the awgn function. The SNR was varied within the range of 0 to 15 dB in increments of 0.5 dB. The received noisy signal was demodulated using the pskdemod function, and the resulting bits were directly compared to the original data to calculate the BER for each SNR value.

5.1.3. Scenario with convolutional coding. To enhance the reliability of the transmitted data, the author incorporated convolutional coding. A convolutional encoder with a polynomial representation defined by poly2trellis(7, [133 171]) was employed. The raw data bits were encoded using this convolutional scheme through MATLAB's convenc function. As with the baseline scenario, the encoded data bits underwent PSK modulation, followed by the introduction of AWGN with varying SNR values. Upon reception, the signal was demodulated and subsequently decoded using the Viterbi algorithm, realized through MATLAB's vitdec function with a truncation depth of 5. The decoded data was then compared to the original set to compute the BER for each SNR setting.

5.1.4. Scenario with Block Coding (Hamming-like 7,3). Lastly, the author explored the utility of block coding, specifically a simplified Hamming code of length 7 with a data block size of 3. Custom functions, named hamming Encoder and hamming Decoder, were crafted to encode and decode data blocks, respectively. The encoding function appends four parity bits to each 3-bit block, while the decoding function retrieves the original 3-bit data from the 7-bit encoded block. This encoded data was then modulated, transmitted through the noisy channel, and demodulated on the receiving end, just like in the prior scenarios. Once received, the data was decoded block-by-block using the hamming Decoder. The resulting BER was determined by contrasting the decoded data blocks with the original ones over varying SNR levels.

This design ensured a thorough comparison between the performances of raw transmission (without coding), convolutional coding, and block coding mechanisms in a simulated satellite communication environment.

5.2. Channel condition-based performance assessment

5.2.1. Introduction. In the quest to examine the influence of different channel conditions on the performance of various encoding techniques, the author specifically focused on the variations in SNR and its ramifications on the overall system's reliability.

5.2.2. SNR variations and their implementation. The author implemented a sequence of SNR values, ranging from 0 dB to 15 dB in increments of 0.5 dB. This range of SNR scenarios was intended to emulate a broad spectrum of potential real-world satellite channel conditions – from highly noisy environments (low SNR values) to relatively clean transmission channels (high SNR values).

DOI: 10.54254/2755-2721/35/20230385

5.2.3. Modulation and noise simulation. For each distinct SNR setting, the author introduced a PSK modulation scheme to modulate the raw binary data for transmission. Following this, the AWGN model was employed to simulate the noisy satellite channel for the specific SNR scenario. This model serves as an essential tool in communications systems analysis, given its ability to replicate the kind of random noise typically encountered in satellite transmissions.

5.2.4. Impact of noise on signal quality. The immediate consequence of changing the SNR level was observed in the demodulated received data. For each SNR setting, the transmitted signals underwent AWGN perturbations, leading to alterations in the received signals.

5.2.5. BER analysis and encoding comparison. The demodulated received data was then decoded, and the BER was calculated by contrasting the original data with the received data.

Three different encoding schemes were assessed: No Coding, Convolutional Coding, Simplified Block Coding (Hamming-like 7,3).

5.2.6. Observations and conclusions. As the SNR value increased, a perceptible decline in the BER was evident, underscoring the direct link between SNR and the error rate. This observation is in line with the expectation that as the transmission signal's strength or clarity improves relative to the noise (i.e., as SNR increases), the system's propensity to make errors diminishes.

Nevertheless, the extent of this BER reduction was distinct for each encoding technique. For instance, while the "No Coding" scenario naturally experienced a higher BER across most SNR values, the convolutional and block coding schemes demonstrated increased resilience to noise, especially at lower SNR levels. This delineation accentuates the significance of employing robust encoding techniques to enhance satellite communication performance, especially in noise-rich environments.

5.3. Comparative evaluation based on encoding schemes

5.3.1. Background. To understand the performance of different encoding techniques in satellite communication, the author embarked on a detailed comparative evaluation. The analysis was anchored on two primary factors: the BER across various encoding rates and the resilience of each encoding rate in the face of noise and interference.

5.3.2. Methodology and setup. Utilizing MATLAB as the simulation platform, the author designed three distinct scenarios: one without any coding, one employing convolutional coding, and a final one using a simplified block coding resembling a Hamming 7,3 code. A SNR was systematically varied from 0 to 15 dB in increments of 0.5 dB to simulate different channel conditions.

No Coding Scenario: In this setup, data was directly modulated using PSK without any preceding encoding. Post modulation, the data was transmitted over an AWGN channel. At the receiver, PSK demodulation was performed and the BER was computed by comparing the received data with the original. The results were stored in the BER_No_Coding array.

Convolutional Coding Scenario: The data was first encoded using a predefined trellis structure, which was then modulated using PSK. After transmitting over the AWGN channel and demodulating at the receiver, the Viterbi algorithm was employed for decoding. The resultant BER for this setup was stored in the BER_Convolutional array.

Simplified Block Coding (Hamming-like 7,3) Scenario: Here, the data was divided into blocks and then encoded based on a modified Hamming mechanism. Similar to the previous setups, the data was then modulated, transmitted, received, demodulated, and subsequently decoded. The BER was calculated and stored in the BER_Block array.

DOI: 10.54254/2755-2721/35/20230385

5.3.3. Results and discussion. From the simulations, distinct BER curves for each encoding technique were plotted against the SNR range. This provided invaluable insights into how each coding strategy responded to noise and interference.

No Coding: As expected, without any form of error correction, the BER for the no-coding scenario was considerably high, especially in environments with a lower SNR.

Convolutional Coding: The convolutional coding technique showcased improved resilience to noise and interference compared to the no-coding scenario. As SNR increased, the BER showed a more pronounced decrease, indicating its efficiency in combating errors.

Simplified Block Coding: The block coding technique, inspired by the Hamming code, demonstrated a performance that was a trade-off between the no-coding and convolutional coding scenarios. Its ability to correct errors was evident, but its performance was inherently tied to the characteristics of the block code design.

5.3.4. Summary. In summary, the author's comparative evaluation illuminated the significant role of encoding schemes in satellite communication, particularly in noisy environments. While no coding offers simplicity, its vulnerability to errors is palpable. Convolutional coding, with its intricate design, excels in error correction, especially at higher SNRs. The block coding approach offers a balanced performance, presenting a viable alternative for specific satellite communication applications.

6. Comprehensive analysis & discussion on experiment outcomes

6.1. Superiority of convolutional coding

As Figure 3 shows, among the three schemes analysed, convolutional coding exhibited the most commendable performance. The BER for convolutional codes consistently stayed below those of the other two strategies across the entire range of the SNR. This can be attributed to the inherent nature of convolutional codes. They operate by adding memory to transmitted data, which means they spread the error correction capability over several bits, allowing for a more resilient transmission. This "memory effect" undoubtedly plays a pivotal role in its superior performance, especially in challenging communication environments like satellite transmissions where signal attenuation and noise can be significant.

DOI: 10.54254/2755-2721/35/20230385

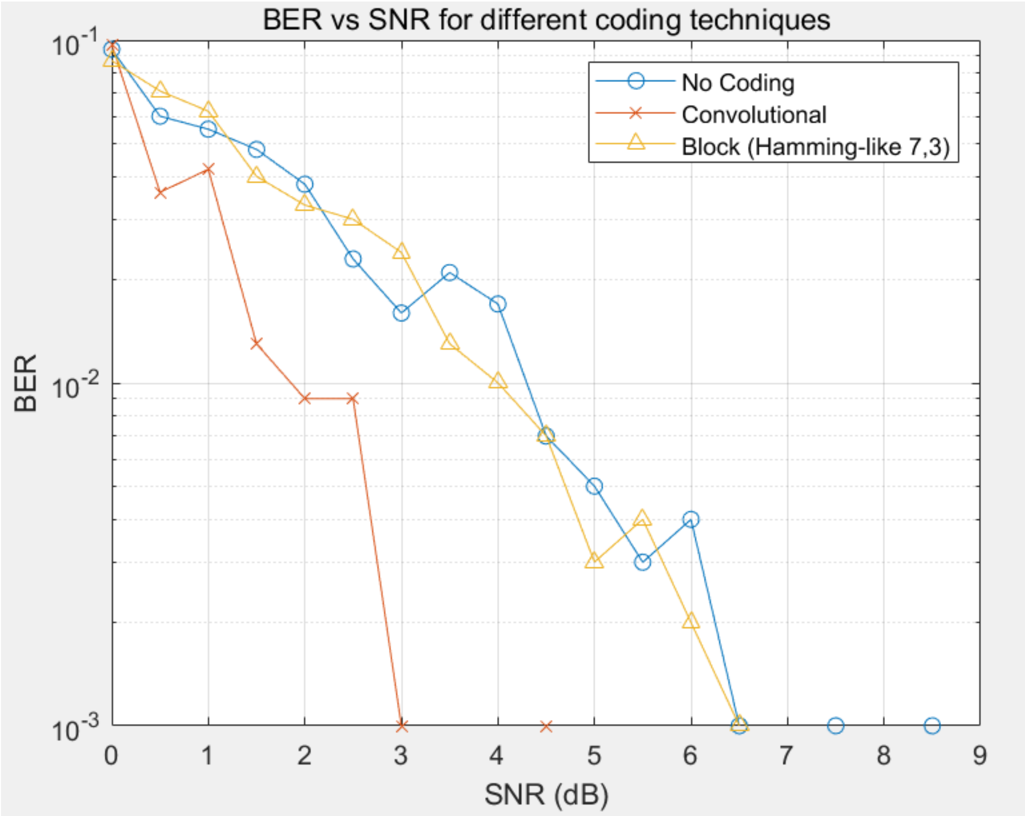

Figure 3. Running result.

6.2. Comparative analysis: no coding vs. block coding

According to Figure 3, when analysing the BER for the 'No Coding' scheme and the 'Block (Hamming-like 7,3)' scheme, the differences between them were not as pronounced as initially anticipated. While the Hamming-like block code did provide some error correction capabilities, its efficiency in doing so was not significantly better than the baseline of no coding. This observation can be reasoned out by considering the rudimentary nature of the Hamming-like 7,3 code used in the simulation. Block codes, especially simple ones like the Hamming, primarily aim to detect and correct errors in individual blocks of data. When juxtaposed against the continuous error correction capability of convolutional codes, it is evident why block codes might fall short in certain scenarios.

6.3. Practical implications for satellite communication

For satellite communication where reliability and data integrity are crucial, the results suggest that convolutional coding might be a more appropriate choice for error correction. However, the selection of a coding scheme should not solely be based on BER performance. Factors such as implementation complexity, required computational power, and latency also play an essential role. While convolutional codes offer superior BER performance, they might be computationally more intensive than block codes. This trade-off needs to be evaluated based on specific use cases and system requirements.

6.4. Potential limitations and avenues for future research

The analysis was based on a simplified block coding scheme, and there are more advanced and intricate block coding methods available. Future studies might consider evaluating more complex block coding techniques against convolutional codes. Moreover, introducing varying channel conditions and other modulation schemes could provide a more comprehensive understanding of these coding strategies' adaptability and performance in real-world satellite communication scenarios.

DOI: 10.54254/2755-2721/35/20230385

7. Conclusion

The comparative analysis of coding techniques in satellite communication demonstrated a clear superiority of convolutional codes in terms of BER performance. Over the range of signal-to-noise ratios tested, convolutional codes consistently achieved a notably lower BER compared to both the no coding baseline and the block coding technique. The difference between the no coding and block coding (Hamming-like 7,3) was marginal, indicating limited advantages of this particular block coding approach. Thus, for applications demanding optimal reliability in satellite communication, convolutional codes emerge as the preferred choice.

References

[1] N. F. N. Suzi Seroja Sarnin, Wan Nor Syafizan W. Muhamad, "Performance Evaluation of Phase Shift Keying Modulation Technique Using BCH Code, Cyclic Code and Hamming Code Through AWGN Channel Model in Communication System," IEEE Transactions on Communications, 2010.

[2] K. J. Hole, "A comparison of trellis modules for binary convolutional codes," IEEE Transactions on Communications, vol. 46, no. 10, pp. 1245-1249, 1998.

[3] R. K. P. P. Dayal, "Implementation of Reed-Solomon CODEC for IEEE 802.16 network using VHDL code," 2014 International Conference on Reliability Optimization and Information Technology (ICROIT), pp. 452-455, 2014.

[4] G. D. Rovera, M. Abgrall, P. Uhrich, P. Defraigne, B. Bertrand, and Ieee, "GNSS antenna multipath effects," in European Frequency and Time Forum (EFTF), Torino, ITALY, Apr 10-12 2018, NEW YORK: Ieee, 2018, pp. 208-212.

[5] R. C. Smith and R. C. Lange, "Signal to noise ratio," (in English), Crit. Rev. Diagn. Imaging, Article vol. 42, no. 2, pp. 135-140, 2001.

[6] K. Yamaguchi and H. Imai, "Construction of unequal error protecting convolutional codes from low rate convolutional codes," (in English), Proceedings. 1994 IEEE International Symposium on Information Theory (Cat. No.94CH3467-8), Conference Paper pp. 275-275, 1994 1994.

[7] P. Loskot and N. C. Beaulieu, "A family of low complexity adaptive binary linear block codes," (in English), IEEE Globecom 2006, Conference Paper pp. 4662-4667.

[8] D. E. Lazic, T. Beth, S. Egner, Ieee, and Ieee, "Constrained capacity of the AWGN channel," in IEEE International Symposium on Information Theory, Mit, Cambridge, Ma, Aug 16-22 1998, NEW YORK: Ieee, 1998, pp. 237-237.

[9] Y. Tanaka, H. Harashima, and H. Miyakawa, "A simple method for improvement of the error probability performance of PSK," (in English), Trans. Inst. Electron. Commun. Eng. Jpn. E, Engl. (Japan), vol. E61, no. 7, pp. 553-554, July 1978.

[10] G. Hori and S. Sagayama, "Variant of Viterbi algorithm based on p-Norm," (in English), 2017 22nd International Conference on Digital Signal Processing (DSP), Conference Paper pp. 5, 2017.

[11] A. Pavlidis, A. Arapoyianni, and D. Loukas, "Fully-digital, highly-modular architecture of a Hamming-like neural network," (in English), 9th European Signal Processing Conference (EUSIPCO 1998), Conference Paper pp. 4, 1998.