International Journal of Innovative Research and Scientific Studies, 8(5) 2025, pages: 245-255

IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

Hyperchaotic map-based adaptive polar codes for robust and efficient communication

Hadjer SOUILLAH[1*], Lahcen HADJ ABDERRAHMANE[2], Abderrahmene HADJ BRAHIM [3], Adda ALI PACHA[4], Belkacem IMINE [5]

[1,3,4,5]*Laboratory of Coding and Security of Information, Faculty of Electrical Engineering, Department of Electronics, University of Sciences and Technology of Oran Mohamed Boudiaf USTO-MB B.P. 1505, El Mnaouar – Bir el Djir, Oran, Algeria.*
[2]*Algerian Space Agency (ASAL), Centre of Satellite Development (CDS) POS 50 Ilot T12 Bir el Djir, Oran, Algeria.*

Corresponding author: Hadjer SOUILLAH (*Email: hadjer.souillah@univ-usto.dz*)

Abstract

This paper introduces a novel chaos-driven adaptive polar coding scheme designed to enhance both the reliability and security of digital communications. The proposed system employs a two-dimensional (2D) hyperchaotic map to dynamically control coding parameters at the block level. For each block, the chaotic system generates two indices: one selects the block length from a predefined set of sizes 2^n, and the other selects a signal-to-noise ratio (SNR) value from a fixed SNR vector of length 1×11. The selected SNR value guides the calculation of the Bhattacharyya parameter, enabling optimized frozen bit selection during polar code construction. A shared secret key, composed of the chaotic map's initial conditions and the SNR vector, ensures that the receiver can accurately regenerate the parameter sequence without any explicit exchange of configuration data. The system is evaluated over an additive white Gaussian noise (AWGN) channel using BPSK modulation. Simulation results demonstrate a clear improvement in bit error rate (BER) performance and enhanced security compared to conventional static polar coding. Furthermore, sensitivity analysis shows that decoding fails completely in the presence of key mismatch, thus ensuring strong data confidentiality. This approach offers an efficient and secure communication framework, particularly suited to dynamic and resource-constrained environments such as satellite systems and wireless sensor networks.

Keywords: AWGN, BER, Bhattacharyya parameter, Hyperchaotic system, Polar codes, Security.

DOI: 10.53894/ijirss.v8i5.8604
Funding: This study received no specific financial support.
History: Received: 2 June 2025 / **Revised:** 7 July 2025 / **Accepted:** 9 July 2025 / **Published:** 17 July 2025

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Publisher: Innovative Research Publishing

1. Introduction

In modern communication systems, particularly in critical infrastructures such as satellite communications, ensuring reliable and secure data transmission is of paramount importance. These systems often operate in harsh and unpredictable environments where signal degradation caused by noise, interference, fading, or even intentional jamming can severely impact performance. Furthermore, with the increasing prevalence of cyber-physical threats and the growing demand for real-time data services, it becomes essential to design transmission schemes that not only correct errors efficiently but also safeguard information against adversarial attacks.

Coding and decoding optimization play a vital role in addressing these challenges by enhancing the robustness and security of communication links. Since the advent of error-correcting codes, various techniques, ranging from classical block codes and convolutional codes to more advanced schemes such as LDPC and turbo codes, have been developed to approach channel capacity while minimizing computational complexity. These approaches aim to balance error performance, throughput, latency, and hardware efficiency.

As communication systems continue to evolve toward higher data rates and increasingly stringent security requirements, particularly in resource-constrained platforms such as satellites or embedded systems, traditional static coding schemes exhibit significant limitations. Fixed parameters are often suboptimal under variable channel conditions and cannot ensure resilience against sophisticated attacks. Consequently, there is a growing demand for adaptive and secure coding strategies capable of dynamically responding to fluctuating transmission environments while maintaining low processing overhead and minimizing information leakage.

In this context, polar codes and chaotic systems present complementary features. Polar codes are recognized for their capacity-achieving properties and efficient decoding algorithms, whereas chaotic maps offer strong unpredictability and dynamic flexibility for secure key generation and parameter adaptation. This synergy motivates the exploration of integrated frameworks that combine chaotic dynamics with adaptive polar code structures to achieve secure and efficient communication.

1.1. Literature Review

The integration of error-correcting codes into cryptographic systems has been an active research area for decades. Among the earliest examples, the McEliece [1] Cryptosystem, proposed in. This system is based on the use of binary Goppa codes and is well-known for its high encryption and decryption speed. However, one of the main drawbacks of this cryptosystem lies in the large size of its public keys, which has motivated extensive research aimed at improving its efficiency. A variant of this scheme, proposed by Niederreiter [2], relies on the parity-check matrix instead of the generator matrix, enabling more efficient encryption while maintaining the robustness of the McEliece [1] system.

In parallel, Rao and Nam [3] proposed a secret-key cryptosystem based on Hamming codes, aiming to reduce key size while maintaining a high information rate. Their approach involves keeping the generator matrix of the code secret, but this scheme has proven to be vulnerable to chosen-plaintext attacks [4].

Polar codes, later introduced by Arikan [5] marked a major breakthrough in the field of channel coding. These codes are the first to be proven capable of achieving the capacity of discrete memoryless channels, while offering efficient decoding algorithms such as successive cancellation (SC) and its improved variants, like successive cancellation list (SCL) decoding [6]. The efficiency of polar codes relies on the identification of frozen bits, which are selected based on the Bhattacharyya parameter, itself dependent on the channel's signal-to-noise ratio (SNR) [7]. Polar codes have recently attracted significant attention in cryptographic applications due to their structured design and capacity-achieving properties [8].

However, conventional approaches suffer from limitations due to the rigidity of code parameters, such as static block sizes and the lack of adaptability to channel conditions. To overcome these issues, several studies have explored the integration of chaotic maps to dynamically generate pseudo-random sequences and make the system more robust and flexible [9, 10]. These maps exhibit complex nonlinear behavior, reducing the system’s vulnerability to structural attacks and enabling dynamic resource allocation [11].

Recent cryptanalysis efforts in polar code-based cryptography have exposed critical vulnerabilities despite proposed structural hardening techniques. The PKC-PC framework [12] for instance, attempts to enhance security through generator matrix obfuscation via coordinate masking (stochastic row selection) are made. However, such schemes may still be vulnerable to attacks exploiting the structure of minimum weight codewords [13] or to attacks based on code equivalence [14].

Another relevant framework is RLCE (Random Linear Code Encryption), proposed by Wang et al. [15]. This framework is inspired from the McEliece [1] cryptosystem aims to enhance security by using a combination of structured and randomized codes to strengthen protection. RLCE is designed employing a distortion matrix with hybrid structured/random column configurations, a design choice intended to resist classical cryptanalytic attacks. However, RLCE was not selected for the second round of the NIST standardization process due to vulnerabilities found in some of its initial parameters [16].

The reviewed literature highlights a persistent trade-off between structural efficiency and cryptographic robustness. While polar codes offer capacity-achieving performance, their static nature limits adaptability and resilience. Chaotic systems introduce flexibility and unpredictability, yet existing integrations with polar codes do not fully exploit block-level dynamic adaptation in both structure and channel conditions.

1.2. Motivation and Contribution

This work proposes a secure and adaptive transmission scheme based on a two-dimensional (2D) chaotic map to enhance the security and robustness of polar codes. Unlike conventional approaches where the code structure is statically defined, our method introduces controlled variability for each transmitted data block.

A fixed code rate of $R = 1/2$ is employed throughout all simulations in this study to enable consistent comparisons across varying block sizes and channel conditions. This choice simplifies the performance analysis while maintaining the flexibility of block segmentation and SNR adaptation.

Specifically, the proposed method relies on the following principles:

- Dynamic segmentation of the message (information bits) into independent blocks of size $N = 2^n$, where each block size is determined by a 2D chaotic sequence.
- Block-wise SNR adaptation, enabling dynamic recalculation of Bhattacharyya probabilities and optimized selection of frozen bits for polar code construction.
- Synchronization between the transmitter and the receiver is achieved by using a shared key that contains the initial conditions of the chaotic system and the selected SNR values.
- Security enhancement through unpredictability: the code parameters and structure change for each block, significantly increasing the complexity of interception or attacks without knowledge of the key.

1.2.1. Key Contributions

The main contributions of this work are as follows:

- The design of an adaptive polar coding system driven by a 2D chaotic map, with per-block dynamic adjustment of code rate and SNR.
- Transparent decoder synchronization, based on local regeneration of the chaotic sequence, allows the receiver to reconstruct the code structure without explicit parameter exchange.
- Robustness evaluation through AWGN channel simulations demonstrates both resistance to disturbances and improved security compared to static systems.
- Sensitivity analysis to chaotic key misalignment, showing that decoding fails rapidly when the key is incorrect, thus ensuring data confidentiality.
- Potential integration into constrained and variable environments, such as satellite communications, where channel conditions frequently change.

1.3. Document Organization

This paper is structured as follows: Section 2 introduces the fundamental concepts of polar codes and outlines the concepts of chaotic maps. Section 3 describes the architecture of the proposed chaos-driven adaptive system, including the methodology for message segmentation, the adaptive parameter selection mechanism, and the design of the cryptographic key structure. In Section 4, we evaluate system performance in terms of BER, based on channel conditions and the selected parameters. A comparative analysis is performed between our dynamic segmentation approach and a static segmentation method from previous works, followed by an assessment of the system's robustness against potential attacks, highlighting its security advantages and complexity. Finally, Section 5 concludes this work and suggests directions for future research.

2. Preliminaries

2.1. Polar Codes

2.1.1. Channel Polarization Principle

Channel polarization is a technique discovered by Arikan [5] which takes a set of N copies of a noisy channel and applies recursive linear transformations to produce N polarized sub-channels, some of which become extremely reliable, while others become extremely noisy. This phenomenon enables polar codes to achieve channel capacity.

2.1.2. Basic Transformation

Polar codes recursively transform channels using the Kronecker transformation matrix. The basic generator matrix is given by:

$$G_2 = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad (1)$$

Through recursive application, the generator matrix for a code of length $N = 2^n$ is given by:

$$G_N = G_2^{\otimes n} \quad (2)$$

Where $\otimes$ denotes the Kronecker product.

2.1.3. Bhattacharyya Parameter

The Bhattacharyya parameter $Z(W)$ measures the reliability of a channel W. It is defined as [5]:

$$Z(W) = \sum_y \sqrt{W(y|0)\, W(y|1)} \quad (3)$$

Where $W(y|x)$ is the output distribution of the channel for an input x.

During channel polarization, a channel W is split into two sub-channels:

- Lower sub-channel W^- (less reliable):

$$Z(W^{-}) = 2Z(W) - Z(W)^2 \Leftrightarrow Z\left(W_N^{(2i-1)}\right) = 2Z\left(W_{N/2}^{(i)}\right) - Z\left(W_{N/2}^{(i)}\right)^2 \quad (4)$$

- Upper sub-channel W^{+} (more reliable):

$$Z(W^{+}) = Z(W)^2 \Leftrightarrow Z\left(W_N^{(2i)}\right) = Z\left(W_{N/2}^{(i)}\right)^2 \quad (5)$$

The initial value of the Bhattacharyya parameter for an AWGN channel with BPSK modulation is given by:

$$Z(W_1^1) = e^{-\frac{RE_b}{N_0}} \quad (6)$$

Where R is the code rate, E_b is the average energy per transmitted bit, and N_0 is the noise power spectral density.
After several transformation stages, some channels become nearly perfect ($Z \approx 0$) while others become almost unusable ($Z \approx 1$). The set of sub-channels with low Bhattacharyya values is selected for transmitting information.

2.1.4. Polar Code Encoding

A polar code is defined by four parameters:

- Code length N**:** always a power of 2 ($N = 2^n$).
- Code rate R**:** defined as $R = K/N$, where K is the number of information bits.
- Frozen bit indices $\mathcal{F}$**:** a set containing the $N - K$ indices of the frozen bits.
- Frozen bits vector $\mathcal{V}_{\mathcal{F}}$**:** a binary vector of length $N - K$, containing the values of the frozen bits (usually all set to 0).

The encoding of a message u into a code word x is given by:

$$x = u \cdot G_N \quad (7)$$

Where $u = (u_1, u_2, \cdots, u_N)$ denotes the input vector composed of the following elements:

- K information bits are placed in the appropriate sub-channels.
- $N - K$ frozen bits, usually fixed to 0.

2.1.5. Polar Code Decoding

Successive Cancellation Decoding (SCD) is the simplest and most fundamental decoding method for polar codes. It leverages the recursive structure of the polar transformation and follows a sequential approach, where each information bit is successively estimated based on previous decisions. The objective of SCD is to recover the input vector from the codeword transmitted over a noisy channel, such as an AWGN channel.
The process follows these steps:

1. At the input of the receiver, y is given by Proakis and Salehi [17]:

$$y = x + n \quad (8)$$

Where:

- $x \in \{+1, -1\}$ is the transmitted signal (BPSK: $x = 1 - 2c$, where c is the encoded bit)
- $n \sim \mathcal{N}(0, \sigma^2)$ is Gaussian noise.

The signal-to-noise ratio (SNR) in terms of energy per bit is:

$$SNR = \frac{E_b}{N_0} \quad (9)$$

The log-likelihood ratio (LLR) is defined as:

$$L(y) = \log \frac{P(x = 0|y)}{P(x = 1|y)} \quad (10)$$

For an AWGN channel with BPSK, we get Proakis and Salehi [17]:

$$L(y) = \frac{2y}{\sigma^2} \quad (11)$$

2. The LLR updates are defined by Balatsoukas-Stimming et al. [18]:

$$L_i^{(j)} = f\left(L_i^{(j-1)}, L_{i+2^{j-1}}^{(j-1)}\right) \quad (12)$$

With:

$$f(L_1, L_2) = 2\tan^{-1}\left(\tan\left(\frac{L_1}{2}\right)\tan\left(\frac{L_2}{2}\right)\right) \quad (13)$$

In addition, the sum update:

$$L_{i+2^{j-1}}^{(j)} = L_{i+2^{j-1}}^{(j-1)} + (-1)^{\hat{u}_i} L_i^{(j-1)} \quad (14)$$

Where $\hat{u}_i$ denotes the estimated value of bit u_i obtained after the decision process**.**

3. The SCD decoder starts by estimating u_1, then u_2, ..., up to u_N. Each bit is successively estimated using the LLR and the previous decisions:

$$\hat{u}_i = \begin{cases} 1, & L_i < 0 \\ 0, & L_i \geq 0 \end{cases} \quad (15)$$

2.2. 2D Hyperchaotic Map and Bifurcation Diagram

The 2D hyperchaotic map is a nonlinear dynamical system that generates chaotic sequences using two state variables, x_n and y_n. Compared to 1D chaotic systems (e.g., the logistic map), this two-dimensional model exhibits more complex and unpredictable behavior, making it particularly useful in chaotic cryptography.
The system is governed by a pair of coupled discrete-time equations, such as:

$$x_{n+1} = f(x_n, y_n) \quad (16)$$
$$y_{n+1} = g(x_n, y_n) \quad (17)$$

Where f and g are nonlinear functions, often based on trigonometric, quadratic, or exponential forms. A typical example is the 2D trigonometric hyperchaotic model [19]:

$$\begin{aligned} x_{n+1} &= sin^2(a\pi x_n + by_n) \\ y_{n+1} &= cos^2(b\pi/y_n + bx_n) \end{aligned} \quad (18)$$

Where:

- a and b are control parameters that influence the chaotic behavior of the system,
- x_n and y_n are the state variables evolving over time? A slight change in the initial conditions and leads to radically different trajectories after several iterations. This total unpredictability is a key characteristic of chaotic systems and ensures that an attacker cannot reconstruct the chaotic sequence without precise knowledge of the initial parameters.

The bifurcation diagram represents the asymptotic values of the state variables and after a large number of iterations as a function of the parameter. In this study, we analyzed the evolution of the chaotic behavior of the 2D hyperchaotic map by varying the control parameter in the range, while keeping constant as illustrated in Figure 1.

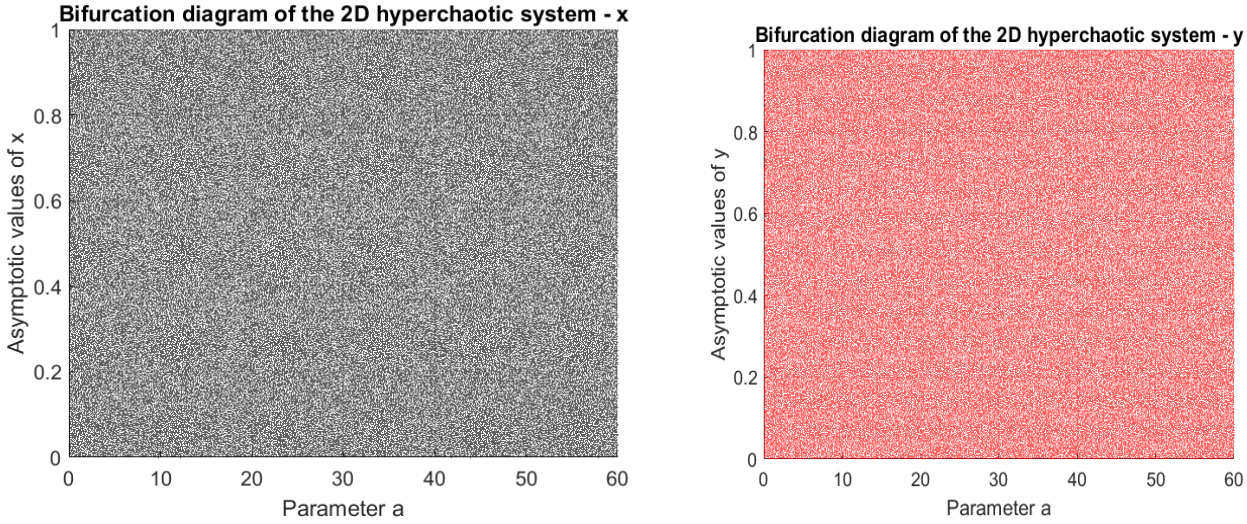

Figure 1.
Bifurcation diagram of the hyperchaotic system (18) depicting the evolution of state variables x and y as control parameter a is varied.

2.3. Methodology: Adaptive Polar Coding with Encryption Behavior Based on a Hyperchaotic Map

In this work, we propose a novel approach to enhancing the security and robustness of digital communications by combining polar codes with a two-dimensional (2D) hyperchaotic map. The system operates based on a shared secret key between the transmitter and the receiver, consisting of:

- An SNR vector (E_b/N_0) of size 1×11, used to dynamically adapt both the coding scheme and the channel simulation.
- Initial conditions of the 2D chaotic map: $a \in [0, 60]$, $b = 30$, $x_0 = 0.2$ and $y_0 = 0.3$.

This key allows synchronization between the transmitter and receiver without the need to explicitly transmit sensitive parameters.

2.4. Chaotic Sequence Generation

The chaotic map is defined by Equation 18 is implemented with precision, rendering approximation-based attacks virtually infeasible. Even a minute deviation in the parameters produces a completely different sequence, ensuring high sensitivity and security.

To guarantee a long, stable, and statistically independent pseudo-random sequence, 5,000,000 iterations are generated. The first 100 values are discarded to eliminate transient effects. The values from and onward are used to drive the encoding process.

2.5. Dynamic Message Segmentation

Unlike conventional fixed-block approaches, our method exploits the hyperchaotic map to dynamically segment the message into blocks of varying sizes, selected from:

$$\text{block\_sizes} = \{2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048\} \quad (19)$$

The information block size N_i for block i is determined from the chaotic component $x(i)$. Specifically, each value $x(i)$ is transformed into a block size index as follows:

$$\text{b\_index} = mod\big(floor(x(i) * 1000), length(\text{block\_sizes})\big) + 1 \quad (20)$$

This enables the dynamic selection of the block size $N_i = \text{block\_sizes[b\_index]}$ from the available powers of two.

This method ensures reproducible variability, minimizes padding (at most one bit), and increases the structural diversity of each transmission.

2.6. Dynamic Block-Wise SNR Selection

In parallel, the chaotic component $y(i)$ is used to dynamically select an SNR value from the shared vector SNR_values according to:

$$\text{S\_index} = mod\big(floor(y(i) * 1000), length(\text{SNR\_values})\big) + 1 \quad (21)$$

This SNR value serves two essential purposes:

- It is used to compute the Bhattacharyya parameters, which determine the frozen and information bit positions in the polar code.
- It defines the noise level added during transmission over the AWGN channel (Equation 8).

2.7. Polar Encoding and Transmission

For each information block of size N_i, the system applies polar encoding with a fixed code rate of $R = 1/2$. As a result, each information block is expanded to a total block length of $N = N_i/R = 2 \times N_i$, which includes both information and frozen bits. The generator matrix G_N, of size $N \times N$, is constructed using Kronecker products. The encoded bits are then modulated using BPSK and transmitted over an AWGN channel with the dynamically selected SNR.

The random variation in both block size and SNR makes each transmission structurally unique, thereby enhancing the confidentiality properties of the system.

2.8. Receiver Decoding and Synchronization

Thanks to the shared key, the receiver can locally regenerate the same chaotic sequence and thereby reconstruct, for each block:

- The block size N_i.
- The corresponding SNR value.
- The frozen bit positions.

The receiver then performs successive cancellation decoding (SCD) for each block. Since no sensitive information is explicitly transmitted, the system is inherently resistant to interception.

2.9. Robustness and Security

The variability introduced by the hyperchaotic map makes the system highly secure, as an attacker would not only need to know the exact SNR assigned to each block but also reconstruct the entire block segmentation and coding structure. This makes any interception or decryption attempt extremely difficult.

Furthermore, this approach enhances system robustness by dynamically adapting the coding process to channel variations, ensuring more reliable transmission. The BER simulation results confirm that the proposed system achieves both efficient and robust communication while reducing its vulnerability to attacks.

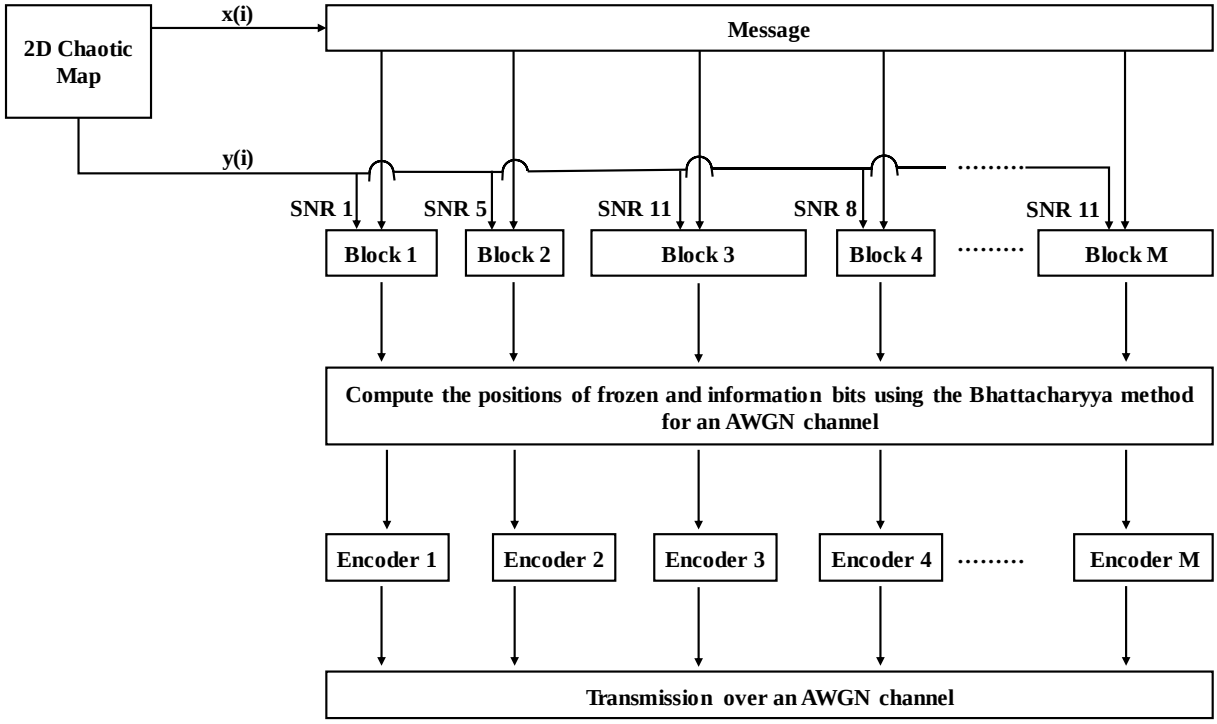

Figure 2.
Secure Data Transmission Architecture Using 2D Chaotic Maps, Polar Codes, and Adaptive Channel Matching.

3. Results and Discussion

In this section, we present the obtained results, highlighting the performance of our approach in terms of bit error rate (BER) and security.

3.1. BER-Based System Performance Analysis

In this phase, we evaluate the system's performance using the bit error rate (BER) metric, considering the dynamic allocation of SNR for each message block. Departing from traditional fixed-SNR methodologies applied uniformly across transmissions, our approach assigns a unique 1×11 pseudo-random SNR vector to each transmission.

Table 1 outlines the SNR configurations applied to six different transmissions and their corresponding post-decoding BER results. Each row corresponds to an independent transmission, demonstrating how the SNR vector directly influences the BER performance.

Table 1.
Comparative analysis of BER performance for six 1 × 11 SNR vectors.

Transmission	SNR Vector	BER
T01	{0.00, 0.50, 1.00, 1.50, 2.00, 2.50, 3.00, 3.50, 4.00, 4.50, 5.00}	2.30×10^{-4}
T02	{0.00, 1.00, 2.00, 3.00, 4.00, 5.00, 6.00, 7.00, 8.00, 9.00, 10.00}	1.80×10^{-4}
T03	{0.22, 1.00, 2.80, 3.25, 4.55, 5.40, 6.06, 7.39, 8.38, 9.28, 10.46}	1.64×10^{-4}
T04	{0.35, 1.55, 2.93, 3.67, 4.72, 6.91, 7.84, 8.10, 8.99, 9.86, 10.95}	1.53×10^{-4}
T05	{1.59, 2.98, 3.83, 4.48, 5.63, 6,91, 7.81, 8.21, 9.58, 10.04, 11.23}	1.18×10^{-4}
T06	{1.85, 2.05, 3,78, 5.11, 5.63, 5.92, 8.14, 8.96, 9.05, 11.05, 11.32 }	1.15×10^{-4}

The results presented in this table show a clear inverse relationship between assigned SNR values and resulting BER. This pattern is consistent with expectations, as a higher signal-to-noise ratio improves the detection of transmitted bits, thereby reducing the error rate.

We observe that SNR vectors with higher average values (e.g., vectors 5 and 6) yield better BER performance. Moreover, even when some blocks are assigned low SNR values (0 to 3 dB), the overall system still maintains a satisfactory BER thanks to the flexibility of dynamic allocation.

By enabling adaptable power distribution at the block level, this dynamic SNR strategy enhances system resilience and ensures optimal performance efficiency while maintaining energy economy.

3.2. Comparative Evaluation of Static versus Dynamic Segmentation Methodologies

This section examines how dynamic block size adaptation *enhances* overall coding efficiency (i.e., improves the coding rate), compared to the limitations of classical static segmentation.

In static approaches, message segmentation introduces unnecessary padding, which decreases coding efficiency.

3.3. Case 1: Static Segmentation (With Padding)

- Suppose a message of 10 000 bits.
- Static segmentation in blocks of 1024 bits, where each block must contain exactly 1024 bits.
- Number of required blocks:

$$\text{Required blocks} = \frac{10\,000}{1024} = 10 \tag{22}$$

- 10 blocks represent $10 \times 1024 = 10\,240$ bits. Hence, 240 padding bits are added.
- Coding rate:

$$R_{fixed} = \frac{10\,000}{10\,240} = 0.9765 \quad (97.65\ \%) \tag{23}$$

3.4. Case 2: Dynamic Segmentation (Padding Reduction)

- The 2D hyperchaotic map selects optimized sizes from the set (2, 4, 8, …, 2048).
- The algorithm adjusts the size to minimize padding. For example:
  - Generated block sizes: 1024, 1024, 2048, 512, 512, 1024, 2048, 512, 512, 512.
  - Total bits transmitted = exactly 10,000 bits (no padding added).
- Coding rate:

$$R_{dynamic} = \frac{10\,000}{10\,000} = 1.0 \ \ (100\%) \tag{24}$$

The improvement in efficiency is given by:

$$\Delta R = \left(\frac{R_{dynamic} - R_{fixed}}{R_{fixed}}\right) \times 100 \tag{25}$$

Substituting the values:

$$\Delta R = \left(\frac{1.0 \ - 0.9765}{0.9765}\right) \times 100 = 2.41\% \tag{26}$$

This analysis shows that our dynamic segmentation approach improves coding efficiency by 2.41% compared to the conventional static-block method. This improvement enables more efficient bandwidth utilization while minimizing padding-related overhead

3.5. Security Analysis of the Proposed System

The proposed system relies on a secret key that combines chaotic parameters with a private SNR vector. Consequently, an attacker's search space is defined by multiple layers of complexity.

3.5.1. Sensitivity Analysis of Initial Conditions

To assess the system's sensitivity to initial conditions, we conducted a controlled test using two nearly identical parameter sets (Key 1 and Key 2) with the following configuration:

- Fixed Parameters: $x_0 = 0.2$, $y_0 = 0.3$, $a = 20$, $b = 40$, with a fixed SNR vector: $\{0.35, 1.55, 2.93, 3.67, 4.72, 6.91, 7.84, 8.10, 8.99, 9.86, 10.95\}$.
- Variable Initial Condition:
  Key 1: $x_0 = 0.2$
  Key 2: x_0=0.200000000000001 (a perturbation of $\sim 10^{-15}$),

Initial Behavior (Iterations 1–7):
The absolute difference $|x_1 - x_2|$ remained negligible ($\sim 10^{-14}$), consistent with the near-identical keys.
Divergence Onset (Iteration 8):
A sharp divergence emerged, with $|x_1 - x_2|$ escalating to values close to 1 (Figure 3). Trajectories rapidly deviated despite the 10^{-14}.
Long-Term Dynamics:
Post-divergence, differences remained large and unstable. This experiment demonstrates:

- Chaotic Sensitivity: The system exhibits exponential sensitivity to infinitesimal changes in initial conditions, a defining feature of chaos.
- Predictability Limits: Even subatomic-scale perturbations 10^{-15}), trigger macroscopic divergence within a few iterations, rendering long-term prediction infeasible.

The abrupt transition from near-identical trajectories to chaotic divergence (visualized in Figure 3) underscores the system's inherent unpredictability under minor parametric variations.

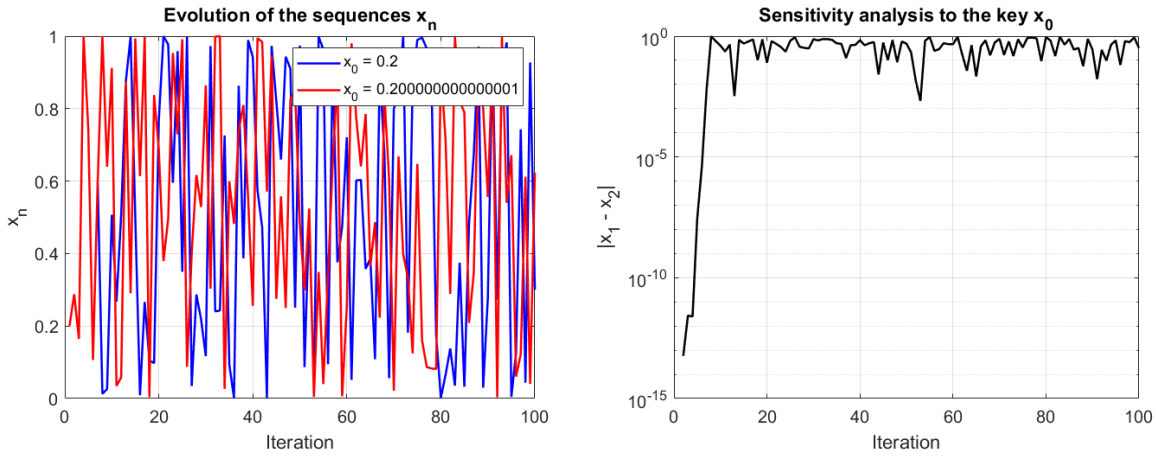

Figure 3.
Comparison of trajectories and their chaotic divergence for two closely related initial keys.

Table 2.
Block Parameters at Transmitter and Receiver Under Perfect Key Synchronization.

Block	Parameters	Transmitter (Key 1)	Receiver (Key 1)
Block 1	Block Size	1024	1024
	SNR_{dB}	8.99	8.99
Block 2	Block Size	16	16
	SNR_{dB}	9.86	9.86
Block 3	Block Size	2048	2048
	SNR_{dB}	9.86	9.86
Block 4	Block Size	2	2
	SNR_{dB}	4.72	4.72
Block 5	Block Size	128	128
	SNR_{dB}	7.84	7.84

Table 3.
Block Parameter Mismatch Between Transmitter and Receiver Due to Key Desynchronization.

Block	Parameters	Transmitter (Key 1)	Receiver (Key 2)
Block 1	Block Size	1024	256
	SNR_{dB}	8.99	10.95
Block 2	Block Size	16	32
	SNR_{dB}	9.86	1.55
Block 3	Block Size	2048	256
	SNR_{dB}	9.86	0.35
Block 4	Block Size	2	512
	SNR_{dB}	4.72	3.67
Block 5	Block Size	128	128
	SNR_{dB}	7.84	8.99

The results presented in Table 2 and Table 3 clearly illustrate the system's high sensitivity to the initial key. Although only a slight change was made to the value of x_0, it is observed that the selected block sizes as well as the assigned SNRs vary from the very first blocks. This rapid divergence demonstrates that the chaotic sequence plays a decisive role in the variability of the coding process, thereby ensuring a highly dynamic and unpredictable structure, an essential property for enhancing the system's security.

Moreover, even in cases where the block size remains identical between two keys, a different SNR results in a change in the polarization parameters of the code. Indeed, as demonstrated in Souillah et al. [20] the positions of the information and frozen bits in polar codes depend directly on the SNR used in the calculation of Bhattacharyya parameters. Therefore, two blocks of the same length but associated with different SNRs will exhibit completely different frozen and information bit positions, which further increases the security and complexity of the coding architecture.

3.6. Chaotic System Parameters

To breach the system, an attacker must first retrieve four critical parameters of the chaotic system: x_0, y_0, a and b, defined with a precision of 10^{-15}. This requirement leads to:

$$\underbrace{10^{15}}_{x_0} \times \underbrace{10^{15}}_{y_0} \times \underbrace{10^{15}}_{a\in[0,60]} \times \underbrace{10^{15}}_{b=30.00} = 10^{60} \approx 2^{200} \quad (27)$$

Our method demonstrates a security level surpassing AES-128, which is generally recognized to provide cryptographic strength equivalent to 2^{128}, making brute-force attack computationally infeasible.

3.6.1. Secret SNR Vector

The SNR vector is a private array containing 11 unique floating-point values, each defined within the interval [0, 12] and specified with a precision of 10^{-15}. The search space associated with each value is:

$$\frac{12}{10^{-15}} = 12 \times 10^{15} \text{ possibilities for each value} \quad (28)$$

Thus, for 11 values:

$$(12 \times 10^{15})^{11} = 12^{11} \times 10^{165} \approx 2^{40} \times 2^{550} = 2^{590} \quad (29)$$

3.6.2. Dynamic SNR Selection Per Block

During transmission, each block dynamically uses one of the 11 secret SNR values. For a total of M blocks, the attacker must consider:

$$11^M \quad \text{additional combinations} \quad (30)$$

3.6.3. Global Complexity for the Attacker

The combined complexity of the entire system can therefore be expressed as:

$$\mathcal{C}_{Total} = 10^{60} \times 12^{11} \times 10^{165} \times 11^M = 12^{11} \times 10^{225} \times 11^M \approx 2^{790} \times 11^M \quad (31)$$

For $M = 1000$:

$$12^{11} \times 10^{225} \times 11^{1000} \approx 10^{11.87} \times 10^{225} \times 10^{1041.39} \approx 10^{1278.26} \approx 2^{4250} \quad (32)$$

This equates to a search space of 2^{4250} possible combinations for an attacker.

Even with the advent of quantum supercomputers under the most favorable projections, such a cryptographic space cannot be fully explored or breached using brute-force or exhaustive methods. The required computational resources would render the task infeasible.

4. Conclusion

In this work, we proposed a novel adaptive approach that integrates polar codes with a two-dimensional hyperchaotic map to simultaneously enhance robustness and security in noisy communication environments. Through dynamic segmentation of the message into variable-sized blocks governed by a hyperchaotic sequence, the system optimizes coding efficiency while increasing structural unpredictability. The secret key, derived from chaotic parameters and a private SNR vector, significantly expands the attacker's search space beyond 2^{4000} Possibilities, thus achieving strong cryptographic resistance. Simulation results validate the scheme's effectiveness, showing superior bit error rate (BER) performance and increased resilience against brute-force and statistical attacks. The proposed scheme offers valuable contributions to mission-critical applications, particularly in environments such as satellite communications and wireless sensor networks, where both reliability and data confidentiality are essential. Its adaptive behavior and lightweight cryptographic design make it suitable for resource-constrained or dynamically varying communication systems, offering a promising framework for secure and efficient transmission. One limitation of the current study is the use of a predefined SNR vector and a fixed code rate $R = 1/2$. While this configuration ensures consistent evaluation, it limits the exploration of rate-adaptive polar coding strategies. Additionally, the scheme assumes perfect synchronization of the chaotic key between sender and receiver, which may present practical implementation challenges in real-world deployments. Future research may explore the extension of this scheme to more complex and realistic channel models, such as Rayleigh or Rician fading environments. Furthermore, the use of higher-order modulation schemes (e.g., QPSK, 16-QAM) could be investigated to assess their suitability for higher-throughput systems. Adaptive SNR selection strategies could also be integrated to further enhance energy efficiency and real-time adaptability in dynamic communication settings.

References

[1] R. J. McEliece, "A public-key cryptosystem based on algebraic coding theory," DSN Progress Report, 1978.

[2] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," *Problems of Control and Information Theory*, vol. 15, no. 2, pp. 159–166, 1986.

[3] T. R. Rao and K.-H. Nam, "Private-key algebraic-code encryptions," *IEEE Transactions on Information Theory*, vol. 35, no. 4, pp. 829-833, 1989.

[4] J. Stern, "A method for finding codewords of small weight," in *Coding Theory and Applications: 3rd International Colloquium Toulon, France, November 2–4, 1988 Proceedings 3*, 1989: Springer, pp. 106-113.

[5] E. Arikan, "Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels," *IEEE Transactions on Information Theory*, vol. 55, no. 7, pp. 3051-3073, 2009.

[6] I. Tal and A. Vardy, "List decoding of polar codes," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2213-2226, 2015.

[7] M. Bardet, V. Dragoi, A. Otmani, and J.-P. Tillich, "On the complexity of computing the Bhattacharyya parameter for polar codes," *IEEE Transactions on Information Theory*, vol. 64, no. 7, pp. 5178–5192, 2018.

[8] B. Imine, R. Saha, M. Conti, and M. Ehsanpour, "Analyzing the potential of polar codes in modern cryptography: A survey," *Archives of Computational Methods in Engineering* 2025. https://doi.org/10.1007/s11831-025-10295-8

[9] G. Kolumbán, M. P. Kennedy, Z. Jákó, and G. Kis, "Chaotic communications with correlator receivers: theory and performance limits," *Proceedings of the IEEE*, vol. 90, no. 5, pp. 711-732, 2002.
[10] L. Kocarev and S. Lian, *Chaos-based cryptography: Theory, algorithms and applications*. Berlin, Germany: Springer, 2011.
[11] X. Wang, Y. Zhang, and M. Wang, "Secure communication using chaotic maps and adaptive encryption," *IEEE Transactions on Circuits and Systems*, vol. 67, no. 12, pp. 4122–4134, 2020.
[12] A. Hooshmand, R. Sadeghi, and F. Yazdanparast, "PKC-PC: A polar code-based public key cryptosystem," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2524–2535, 2022.
[13] R. Misoczki, J.-P. Tillich, N. Sendrier, and P. S. L. M. Barreto, "Reducing key size of McEliece cryptosystem using quasi-cyclic codes," in *Proceedings of the Post-Quantum Cryptography Conference (pp. 15–29)*, 2013.
[14] T. P. Berger and P. Loidreau, "How to mask the structure of codes for a cryptographic use," *Designs, Codes and Cryptography*, vol. 35, pp. 63-79, 2005.
[15] H. Wang, Z. Liu, and K. Qin, "RLCE: Random linear code encryption for post-quantum cryptography," in *Proceedings of the NIST PQC Conference*, 2019.
[16] C. Guo, Y. Lu, and X. Chen, "Cryptanalysis of RLCE," *IEEE Transactions on Information Theory* vol. 67, no. 12, pp. 8196–8205, 2021.
[17] J. G. Proakis and M. Salehi, *Digital communications*, 5th ed. New York: McGraw-Hill, 2008.
[18] A. Balatsoukas-Stimming, M. B. Parizi, and A. Burg, "LLR-based successive cancellation list decoding of polar codes," *IEEE Transactions on Signal Processing*, vol. 63, no. 19, pp. 5165-5179, 2015. https://doi.org/10.1109/TSP.2015.2439211
[19] M. Liu, C. Ning, and C. Zhu, "A Secure Image Encryption Scheme Based on a New Hyperchaotic System and 2D Compressed Sensing," *Entropy*, vol. 26, no. 7, p. 603, 2024. https://doi.org/10.3390/e26070603
[20] H. Souillah, L. H. Abderrahmane, B. Imine, and A. Ali-Pacha, "Effect study of the probability of error on the frozen bit positions in polar codes," *Studies in Engineering and Exact Sciences*, vol. 5, no. 2, p. e12084, 2024. https://doi.org/10.54021/seesv5n2-779