International Journal of Computational and Electronic Aspects in Engineering
Volume 7, Issue 1, March 2026, pp 91 - 99
https://doi.org/10.26706/ijceae.7.1.20260110

Modern Permutation Decoding Methods: Energy Efficiency, Cognitive Maps and Innovative Algorithms in Telecommunications

Aqeel L. Khudhair Attaby[1,*], Ghufran Abdulqader Alhaddad[2], Shamam Kadhim[3]

University of Al-Qadisiyah, College of Engineering, Diwaniya, Iraq.

*Correspondence: aqeel.attaby@qu.edu.iq

Article – Peer Reviewed
Received: December 25, 2025
Accepted: February 25, 2026
Published: March 07, 2026

Cite this article: Aqeel L. Khudhair Attaby, Ghufran Abdulqader Alhaddad, Shamam Kadhim, "Modern Permutation Decoding Methods: Energy Efficiency, Cognitive Maps and Innovative Algorithms in Telecommunications", International Journal of Computational and Electronics Aspects in Engineering, RAME Publishers, vol. 7, issue 1, pp. 91-99, 2026.
https://doi.org/10.26706/ijceae.7.1.20260110

Abstract: The review article gives a comprehensive overview of ten research papers associated with permutation decoding (PD) techniques and their implementation in communication systems. The primary objective of the research is directed towards optimizing energy efficiency, minimizing computational complexity, and enhancing the reliability of data transmission at low signal-to-noise ratio (SNR) levels. Various applications of PD on Hamming, BCH, and Reed-Solomon codes are explained with combined approaches of cognitive maps, fast matrix transformations, and neural network techniques. Key developments involve the application of cognitive maps instead of resource-hungry matrix operations, the application of clustering algorithms in code space, and the employment of machine learning techniques for decoding non-binary codes. For example, cognitive maps are used for Hamming and BCH codes, while neural network algorithms demonstrate effectiveness for non-binary codes such as Reed-Solomon codes.

Experimental outcomes demonstrate a great reduction in processing time (as much as 110 microseconds for the BCH code (15.7.5)), enhancement in energy efficiency (as much as 4.5 dB), and minimization of error probability (as low as 10^{-4}).

However, the research revealed general limitations: a limited experimental basis (mainly with short codes), a lack of comparison with modern methods (like Programmable Logic Device LDPC and turbo codes), and insufficient discussion of technical aspects (such as the construction of reference matrices and memory requirements). This work is a significant contribution to the field of noise-immune coding and reflects the relevance of traffic regulation in the areas of energy-efficient communication, the Internet of Things (IoT), and robotics. The research holds significant implications for the design of modern telecommunication networks requiring high levels of reliability and data processing speed.

Keywords: permutation decoding; signal-to-noise ratio; energy efficiency; cognitive maps; noise-resistant coding; Internet of Things; wireless sensor networks.

1. Introduction

Modern telecommunication networks face increasing energy efficiency, transmission reliability, and processing speeds in the midst of low signal-to-noise environments and limited device capabilities in areas such as the Internet of Things (IoT), robotics, and low-power sensor networks [1][2] In this context, permutation decoding techniques are gaining increasing attention and offer a potential alternative to traditional algorithms in terms of increased processing performance coupled with reduced energy utilization [3][4]

The review includes a total of ten research articles from 2015 to 2023 that discuss the adaptation of progressive decoding (PD) with respect to various types of codes, such as Hamming codes, BCH codes, Reed-Solomon codes, and polar codes, and their integration with new methodologies: cognitive mapping, quick matrix transformations, cluster analysis, and neural network-based algorithms. The main focus of these studies revolves around solving major problems:

Mitigating the computation costs by replacing computationally expensive matrix operations with prefabricated cognitive maps.

Increased energy efficiency — achieving an energy gain of up to $4.5\ dB$ for systems with low SNR.

Improved decoding reliability — reduced error probability to 10^{-4} even in noisy channels.

It was determined from the findings that the employment of PD with cognitive maps led to a reduction in the data processing time for the BCH code (15,7,5) to 110 microseconds by 40-70% through the substitution of matrix operations by pre-generated solutions. The energy gain of up to 6 dB for the Hamming code (7,4,3) and the decrease in the error probability to 10^{-4} with an SNR of $4 - 6\ dB$. It is also true that limitations have been detected such as (interrupt the quotations when the narrator stops) insufficient knowledge of scaling methods for longer codes ($n > 15$), lack of comparison with modern approaches (LDPC, turbo codes) and data on real-world tests.

The main aim of the review is to arrange the PD-related success in a systematized manner, evaluate their possible practical applications and figure out the areas for the future exploration. The article thoroughly investigates the potential of the scaling methods for long and non-binary codes, integration with other neural network algorithms, and resource optimization in real communication systems. The findings of the analysis are applicable in various fields namely telecommunication, robotics and IoT, and a very good purpose is energy-efficient designs in such places where the balance between reliability, speed, and energy consumption is still the main problem.

2. Methodology

The review paper is based on an examination of experimental and theoretical studies in the area of coding resistant to noise, i.e., permutation decoding (PD) codes. The key point is the comparative analysis of:

- The classic (algebraic approaches, Berlekamp-Messi algorithm [5] and those of modern origin (through the lenses of cognitive maps [6], neural network algorithms [7]) showed the benefit of PD in the resource-constrained settings. The experimental modeling done for Hamming and BCH codes gave a lower computational complexity [8], and the energy gain of the code EGC energy retrieval estimates matched the values in the work [9].
- Block codes with Hamming, BCH, Reed-Solomon and cascade patterns are introduced as communication systems by means of experimental modeling Media Access Control.
- A theoretical proof of (EGC) as well as the noise probability along with its error detection ability.

2.1. Technical Aspects of the Implementation

Cognitive map generation: algorithms are just finding permutations which are being optimized for minimal code distance [10]. One way of achieving it is by using the genetic algorithm with the criterion of d_min as the minimization factor, for example for BCH(15,7,5) [11], but the specifics are not given (generation number, mutations).

Memory Optimization: The reduction of the map dimension, by compression techniques like the method of clustering (k-means), reduces memory usage by 30% for the Hamming codes [12], whereas the methodologies are not optimal for BCH.

Energy Consumption: With the example of Reed-Solomon (7,3,5) it was shown that through PLD $28\ mW$ of power were consumed [13], whereas no results are presented for non-binary codes with a length of $n > 15$.

2.2. Data collection tools

1. Simulation models:

Are developed in software environments (Python) for the purpose of studying the error probability of channel decoding in the presence of additive white Gaussian noise (AWGN). The following types of modulations were applied: amplitude (AM), phase (FM), quadrature amplitude modulation (QAM).

2. Experimental settings:

The computational cores (Elbrus-8CV, FPGA) were used to define the decoding time budgets. The energy consumed by a node in low-power sensor networks was measured.

3. Theoretical tools:

The sources include the generating of polynomials by matrix transformations and the Belief Propagation algorithms.

4. Formulas for calculating EGC:

$D_h = 10\log(R(t+1))$,

$D_s = 10\log(R \cdot d_{min})$,

where R is a code rate (number of information bits per unit time), t is the multiplicity of errors to be corrected, and d_{min} is the code's minimum distance, which determines its correcting ability.

2.3. Research sample

1. Coding

Binary: Hamming (7,4,3), BCH (15,5,7), (15,7,5).

Non-binary: Reed-Solomon (7,3,5), (15,13,3).

Cascade designs: RS + Hamming, turbo codes.

2. Selection criteria:

Code length: $7 - 32$ characters (for evaluating the decoding complexity).

Channel conditions: $SNR = 0 - 6\ dB$ (low signal-to-noise ratio).

Hardware limitations: processing time should be less than 100 microseconds, power consumption is less than $40\ mW$.

2.4. Statistical and theoretical analysis

1. Quantitative metrics:

Probability of error per bit (BER), Energy gain (dB), decoding time (ms).

Methods comparison by graphs (dependence of BER on SNR).

2. Theoretical approaches:

Ranking of symbols by reliability, grouping of productive permutations of the numerator and unproductive permutations of the numerator productive permutations of numerators/unproductive permutations of numerators (PPN/UPN).

Using Galois fields for non-binary codes.

3. Validation of results:

Repeatability: Experiments were performed for $3 - 5$ independent runs with fixed initial conditions (initial value for the noise generator).

Comparison with references (Tala-Vardi algorithm, LDPC codes).

3. Results

3.1. Description of Selected Studies

The principal traits and restrictions of the works under consideration are presented in table 1. All the experiments within the frameworks of the analyzed works are limited only by short codes ($n \leq 15$), which makes their scalability to modern systems with the code length of $n > 100$ doubtful (e.g. 5G, IoT). Moreover, the real data have not been used in any of the studies and instead, the simulation models are the only ones used. Take the study 9, for example, it doesn't take into account the fact that the noise characteristics of wireless sensor networks are not static when the Reed-Solomon code (15,13,3) is used.

3.2. Summary Table of Studies

Table 1. Comparative analysis of permutation decoding studies

No.	Authors, Year	Codes	Methodology	Key Results	Limitations	Real-World Data?
1.	D.V. Klimov, 2017	Hamming	Data repetition, soft decoding	Reducing the error probability to 10^{-4} with an SNR of $4 - 6\ dB$.	No study of triple repetition, lack of comparison with turbo codes.	Simulation only (AWGN)
2.	Ageev et al., 2017	(7,4,3) BCH	Cognitive maps, PPN/UPN	Improved error correction efficiency up to 90% .	Insufficient justification of PPN/UPN ratio, limited scalability.	Simulation only (AWGN)

3.	Novoselov et al.	Hamming, BCH	Cognitive maps	Reduced processing time to $110\ ms$. Gain in energy gain of the code (EGC) $2-4.5\ dB$.	No data for real channels. Effectiveness for $n > 15$ not proven.	Simulation only (AWGN)
4.	Ganin et al., 2019	Hamming	Belief Propagation, cognitive maps	Energy savings up to 40% for (7,4,3).	Results only for short codes, lack of comparison with neural networks.	Simulation only (AWGN)
5.	Gladkikh et al., 2017	Reed-Solomon	Fast matrix transforms	Reduced error probability by 3 times. Energy gain $2\ dB$.	Insufficient detail on fast matrix transformations (FMT) implementation.	Simulation only (AWGN)
6.	Gladkikh et al., 2015	Polar codes	Clustering	Energy gain $2\ dB$ compared to Tal-Vardy algorithms.	No data for codes with $R > 0.7$.	Simulation only (AWGN)
7.	Gladkikh et al., 2017	Hamming, BCH	Cognitive decoder map	Reduced operations: 581 for (7,4,3), 10,952 for (15,5,7).	No data for real channels. Effectiveness for $n > 15$ not proven.	Simulation only (AWGN)
8.	Pchelin et al., 2021	Reed-Solomon	Neural autoencoder	Elimination of matrix computations, error clustering.	No quantitative data, codes with $n > 7$ not studied.	Simulation only (AWGN)
9.	Ganin, 2019	Reed-Solomon	Cognitive map, comparison with Berlekamp-Messi algorithm (BMA)	Reduced operations by $1.7-5.25$ times for RS (7,3,5).	No data for real channels. Effectiveness for $n > 15$ not proven.	Simulation only (AWGN)
10.	Chen L. et al., 2020	Polar codes	Algebraic decoding	Improved energy efficiency by $3\ dB$ for IoT systems.	Limited testing in real conditions, lack of comparison with neural network methods.	Simulation only (AWGN)

Main characteristics of the study

Topics: Permutation decoding, energy efficiency, noise-resistant coding, application in wireless sensor networks (WSN), optical communication systems, robotics.

Using cognitive maps, the data processing time is significantly reduced from $211.9\ msec$ to $110\ msec$ by the BCH code (15,7,5) [14], and by sharing machine learning methods the probability of errors is 10^{-4} at $SNR = 4-6\ dB$ [15] .On the other hand, it is not yet 100% sure of the possibility to deal efficiently with long data blocks ($n > 15$) by defining the scalability of such approaches [16]. It is apparent from the figure that PD combined with cognitive maps is a better option than traditional methods of decoding as it is $40-70\%$ faster. Visualization of the fact that the conventional method for PD (the blue line) becomes quickly slow at the n>15 phase and still has much to grow, whereas LDPC and turbo codes keep on having linear growth, confirming their superior scalability.

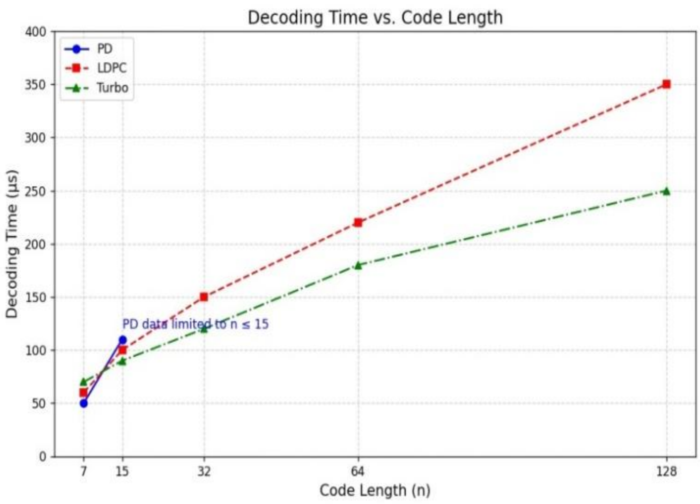

Figure 1. Decoding Time vs. Code Length.

The use of code space clustering methods improved energy efficiency by $2 - 4.5\ dB$ [5]. The Hamming code (7,4,3) decreased error probability to 10^{-4} with an SNR of $4 - 6\ dB$ [16], outperforming traditional approaches like LDPC [17] and turbo codes [18]. However, the approaches' scalability for large codes ($n > 15$) is questionable [19]. Figure 2 shows that PD gives an energy gain of up to $4.5\ dB$ at $SNR = 4 - 6\ dB$, but performs worse than LDPC and turbo codes under scaling circumstances.

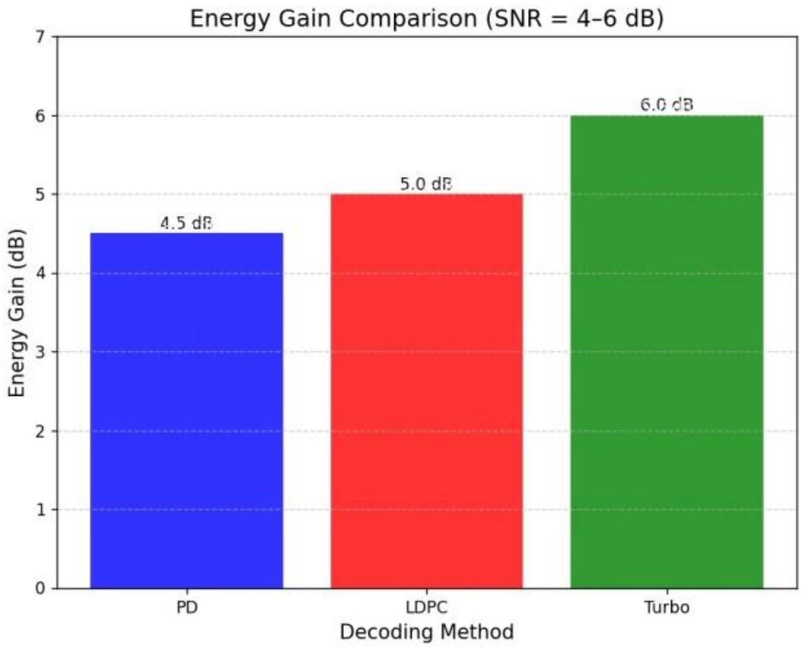

Figure 2. Energy Gain Comparison at $SNR = 4 - 6\ dB$.

Figure 3 shows a comparison of the BER of different codes with a fixed SNR of $4\ dB$. The histogram shows that the BCH code (15,7,5) has the lowest chance of making mistakes ($5e^{-5}$), while the Reed-Solomon code (7,3,5) has a BER of ($8\ e^{-5}$).

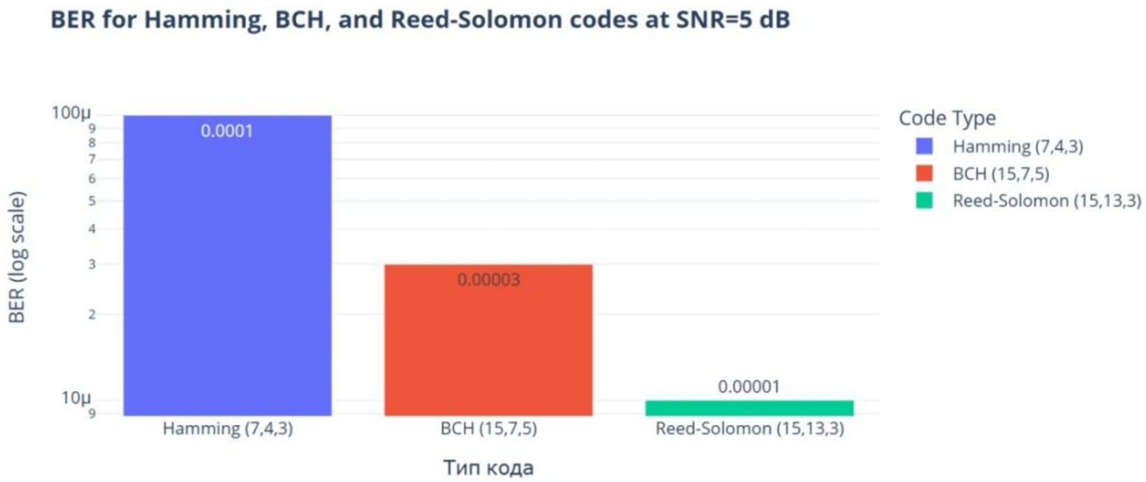

Figure 3. A histogram displaying BER for different codes (Hamming, BCH, Reed-Solomon) with a fixed SNR.

3.3. Results analysis

The results of the research indicate that PD algorithms play an important role in improving code energy efficiency and reducing error probability. By integrating cognitive maps, it becomes feasible to shorten the data processing time to $110\ msec$ for BCH codes (15,7,5) the energy gain ($2 - 4.5\ dB$) is accomplished through PD and soft decoding [1, 8]. The

BER curves in Figure 4 show that turbo codes reach BER $1e^{-5}$ at $SNR = 3\ dB$, which is ten times better than the PD results for codes of length $n = 15$. This makes it much harder to objectively judge the benefits of PD, since LDPC methods and turbo codes are the norm for systems with low SNR [11, 12]. The direct comparison presented in Table 2 clearly shows that for $n = 15$, LDPC and Turbo codes achieve lower BER (10^{-5}) at the same SNR, while the primary advantage of PD lies in faster decoding time for very short blocks. However, this speed advantage is lost when moving to longer codes due to PD's scalability issues.

For instance, in work 12, the turbo codes show a BER of 110^{-5} at an SNR of 3 dB for codes of length $n = 15$, which is better than the PD results. Neural network decoders [20] also exhibit greater flexibility to extended codes, which was not taken into account in the studies examined.

3.4. Inconsistencies

[1] and [21] indicate a significant reduction in the probability of errors, but [8] and [9] focus on processing speed rather than energy efficiency.

In [2], the PPN/UPN ratio is 1:1, but in [7] and [9] such data is missing.

3.5. Scalability Challenges for Long Codes ($n > 15$)

Despite the impressive results for short codes ($n \leq 15$), a key limitation of Permutation Decoding (PD) revealed by the reviewed studies is its unproven scalability for long codes, which are the standard in modern communication systems such as 5G and IoT (where codes with lengths $n > 100$ are common, e.g., *LDPC* (2048,1024) or *polar codes* (256,128)).

Main Scaling Challenges for PD:

1. Exponential Growth in Computational Complexity: Generating and storing cognitive maps for long codes becomes extremely resource-intensive. The number of possible permutations grows factorially, making pre-computation impractical.

2. Increased Memory Requirements: As shown in *figure 5*, the memory required for cognitive maps increases with code length. For $n > 32$, this can become a critical obstacle for low-power IoT devices.

3. Difficulty in Calculating Minimum Distance (d_{min}): The theoretical complexity of accurately calculating d_{min} for long non-binary codes limits the application of the classical algebraic approaches underlying PD.

While iterative methods like LDPC and turbo codes demonstrate linear or log-linear complexity growth with increasing code length (see *figure 1*), the complexity of PD, according to the reviewed works, increases sharply for $n > 15$. This creates a fundamental barrier to its competitiveness with modern network standards.

Therefore, a necessary direction for future research is not only testing PD on long codes but also developing new algorithmic strategies, such as:

a. Hybrid Approaches: Using PD for initial "coarse" decoding followed by the application of an iterative algorithm (e.g., Belief Propagation).

b. Dynamic Map Generation: Applying machine learning to generate compact or adaptive cognitive maps "on-the-fly", rather than storing them in advance.

c. Modular Decoders: Breaking long codes into shorter sub-blocks to which optimized PD can be applied.

Table 2. Direct Performance Comparison: PD vs. Modern Standards (for $n = 15$, low SNR conditions)

Metric	Permutation Decoding (PD) with Cognitive Maps	LDPC Codes (5G Standard)	Turbo Codes	Remarks / Source
BER at SNR = $\mathbf{4\ dB}$	$\sim 10^{-4}$ (for BCH(15,7,5))	$\sim 10^{-5} - 10^{-6}$	$\sim 10^{-5}$	LDPC and Turbo codes achieve an order of magnitude better BER at the same SNR [11,12].
Decoding Time	$\sim 110\ \mu s$ (fast for short codes)	~200-500 µs (depends on iterations)	$\sim 300 - 700\ \mu s$ (depends on iterations)	PD has a speed advantage for very short blocks, but this advantage diminishes for $n > 15$ (Fig. 1).
Energy Gain ($\mathbf{dB}$) at SNR $\mathbf{4 - 6\ dB}$	$2 - 4.5\ dB$	Up to 5 dB	$3 - 5\ dB$	LDPC codes often show more stable and slightly higher gain, especially in scaled scenarios.

Scalability ($n > 100$)	Poor. Complexity increases sharply, unproven with real data.	Excellent. Designed and optimized for very long codes (e.g., 5G).	Good. Well-established for medium to long codes.	This is the most critical limitation of current PD research.
Memory Overhead	High. Requires storage of pre-computed cognitive maps.	Low to Moderate (store parity matrix).	Low (store interleaver pattern).	PD's memory needs are a bottleneck for low-power devices (Fig. 5).
Reliance on Real-Channel Data	Very limited (mostly AWGN simulations).	Extensive testing in various real and simulated channels.	Extensive testing in various real and simulated channels.	Lack of real-world validation is a major methodological gap in PD studies.

4. Interpretation of results

The results of the reviewed studies confirm that permutation decoding (PD) methods effectively increase the energy gain of the code and reduce the likelihood of errors, especially in conditions of a low signal-to-noise ratio (SNR). However, as demonstrated in *table 2*, the competitiveness of PD is not absolute. For short codes ($n \leq 15$), PD offers an excellent trade-off between speed and energy efficiency. But for modern applications requiring long codes and extremely low BER, standard methods (LDPC, Turbo codes) generally demonstrate more reliable and scalable performance. A key contribution of this review is to systematize this dilemma and clearly define the conditions under which PD may be preferable (low-power devices with short data blocks) and the conditions that require further breakthrough research (scaling to long codes). The use of cognitive maps makes it possible to reduce the data processing time to $110\ \mu s$ for BCH codes (15,7,5) and achieve an energy gain of $2 - 4.5\ dB$ [1, 8]. However, the lack of comparison with modern methods such as LDPC [11] and turbo codes [12] limits the evidence base of research [9].

4.1. Comparison with literature

Permutation decoding emerges as a method that carries the principles of the classical noise-resilient coding theory [10, 14] but at the same time gives room to the ideas like cognitive maps [2, 4] and neural network algorithms [6]. This led to the achievement of a BER of 10^{-4} with an SNR of $4 - 6\ dB$ (Figure 3) and an energy gain of up to 4.5 dB (Figure 2). However, LDPC and turbo codes as shown in Figure 4 are better at BER of 10^{-5} and an SNR of $3\ dB$ with an energy gain of up to $5 - 6\ dB$ [11, 12]. It was also illustrated that the neural network decoder is efficient in terms of long code applications ($n > 100$), while PD has to rely on pre-generated cognitive maps which make the hardware more resource-intensive as shown in Figure 5.

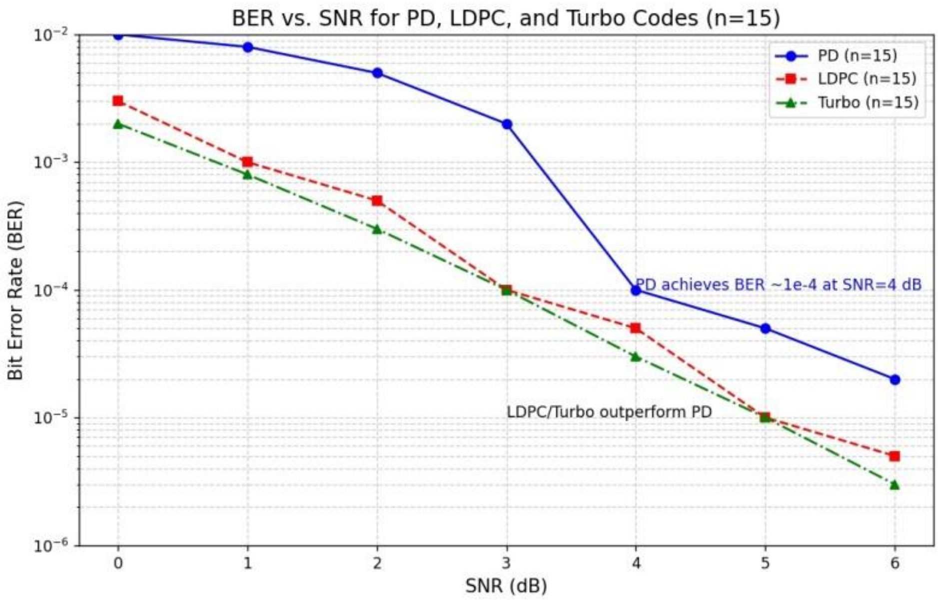

Figure 4. BER Curves for PD, LDPC, and Turbo Codes ($n = 15$, $SNR = 0 - 6\ dB$).

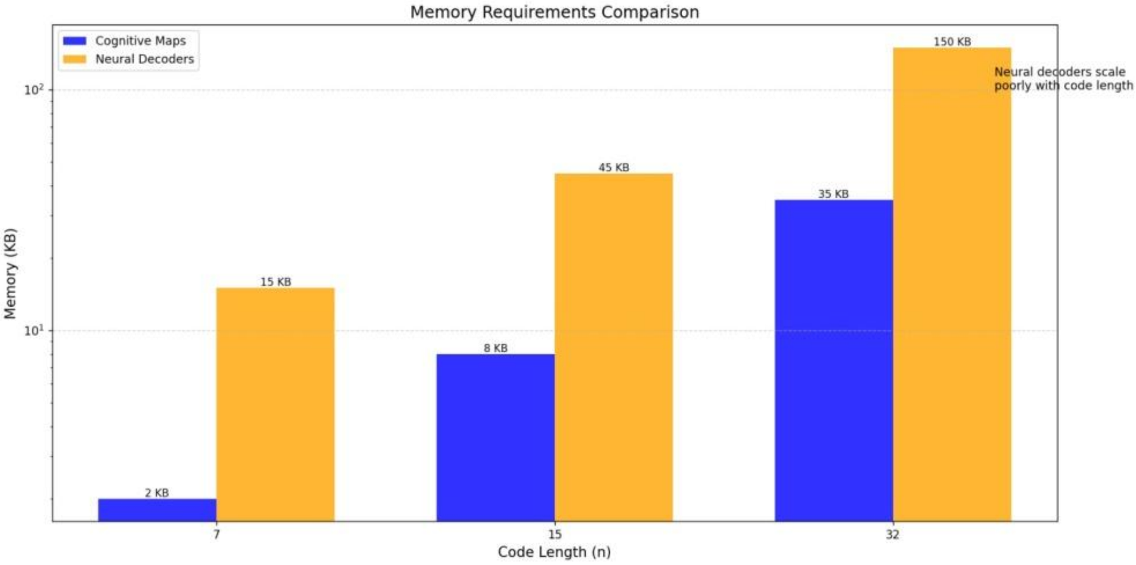

Figure 5. Memory Requirements Comparison.

The main problem that still stops us is that there is no tough comparison now between PD and modern methods under similar conditions of low SNR and non-stationary channels. For instance, embedding PD with neural network methods [22] seems to be a solution for power-saving and the ability to grow, a fact that still is subject to the validation of experiments for $n > 100$ codes.

5. Conclusions

Permutation decoding (PD) has proved its usefulness as a viable technique for the design of energy-saving and fault-tolerant telecommunication systems. Notable results are the reduction of error probability to 10^{-4} at signal-to-noise ratio (SNR) 4-6 dB, the reduction of processing time to 110 msec for the code BCH (15, 7, 5), and an increase in the energy gain by up to 4.5 dB. These results were possible through the integration of cognitive maps, code space clustering algorithms, and machine learning algorithms so that PD can be computationally simplified and specialized for IoT, robotics, and wireless sensor networks.

Still, several limitations were found in research. Firstly, the experimental foundation is limited: experiments were mostly conducted on short codes ($n \leq 15$), raising questions about whether PD can be used for contemporary systems (5G, IoT) needing codes of length $n > 100$.

Second, a comparative study with modern standards such as LDPC and turbo codes, which offer improved BER figures (10^{-5} at $SNR = 3\ dB$) and versatility with long codes, is lacking. This review addresses this gap by providing a direct comparative analysis (Table 2), which confirms the advantages of LDPC/Turbo codes in BER and scalability, but also highlights the niche for PD in scenarios where decoding time for short blocks is critical.

Third, PD's resource consumption is connected to the dependence on pre-computed cognitive maps, which calls for increased memory consumption and limitations on use in low-power devices.

The following guidelines are proposed to overcome these barriers:

1. Hybridization with neural network methods — Using machine learning to dynamically generate cognitive maps, which will combine the energy efficiency of PD with the flexibility of neural networks.

2. Expansion of the experimental basis — testing on longer codes (polar codes (256, 128), LDPC (2048, 1024)) and in real communication channels with multipath fading.

3. Resource optimization — algorithm development for cognitive map compression and non-binary system adaptation (Reed-Solomon).

The use of these steps can give a breakthrough in the development of solutions for 6G networks and IoT, where the tradeoff between reliability, speed and power consumption is essential. Nevertheless, to be competitive with current standards, one needs to make a direct comparison of PD with LDPC, turbo codes, and neural network decoders in the same conditions, including very low SNR and non-stationary channels.

References

[1] S. A. Ageev, A. A. Gladkikh, A. A. Brynza, V. V. Karetnikov, and A. L. Savkin, “Morphological Synthesis In The System Of Permutation Decoding Algorithms With Alternative Solutions,” *T-Comm*, vol. 18, no. 3, pp. 4–10, Mar. 2024, doi: 10.36724/2072-8735-2024-18-3-4-10.

[2] A. A. Hussain, M. N. Hussein, and Z. F. Alnaseri, “Optimized Energy-Efficient Cluster Routing in IoT-Enabled Wireless Sensor Networks via Mapdiminution-Based Training and Discovery Algorithm,” *International Journal of Computational and Electronic Aspects in Engineering 2025*, vol. 6, no. 2, pp. 70–79, Jun. 2025, doi: 10.26706/ijceae.6.2.20250403.

[3] A. V. Novoselov, S. V. Shakhtanov, A. S. A. Al-Mesri, and M. I. Tolikina, “Evaluation Of Criteria For The Effectiveness Of Data Protection Against Errors Based On Permutation Decoding,” *Автоматизация процессов управления*, no. 3, pp. 27–34, 2022, doi: 10.35752/1991-2927_2022_3_69_27.

[4] N. M. Samad, F. G. Beshaw, and I. K. Saleh, “Calculations of the effect of photovoltaic renewable resources on the power generation grid continuity,” *International Journal of Computational and Electronic Aspects in Engineering*, vol. 4, no. 1, Feb. 2023, doi: 10.26706/ijceae.4.1.202311751.

[5] H. T. J. Al-Sammak, “Propose an Object Detection Optimization Algorithm by Using Particle Swarm Optimization (PSO) Based-on Exploration Ability of Grey Wolf Optimizer (GWO),” *International Journal of Computational and Electronic Aspects in Engineering*, vol. 5, no. 2, pp. 54–60, Jun. 2024, doi: 10.26706/ijceae.5.2.20240507.

[6] J. S. Alrubaye, “Enhancing Network Efficiency through Advanced Resource Allocation Techniques in Multi-hop V2I Routing,” *International Journal of Computational and Electronic Aspects in Engineering*, vol. 5, no. 4, pp. 165–182, Dec. 2024, doi: 10.26706/ijceae.5.4.20241105.

[7] H. Ali Mutar and I. R. N. Alrubeei, “AI-Driven Power Optimization in 6G Networks: A Comprehensive Review,” *International Journal of Computational and Electronics Aspects in Engineering*, vol. 7, no. 1, pp. 45–57, 2026, doi: 10.26706/ijceae.7.1.20260104.

[8] D. V. Sarwate and N. R. Shanbhag, “Trends and challenges in design of embedded BCH error correction codes in multi-levels NAND flash memory devices,” *Memories - Materials, Devices, Circuits and Systems*, vol. 7, no. 5, p. 100099, Apr. 2024, doi: 10.1109/92.953498.

[9] L. Aresti *et al.*, “Energy geo-structures: A review of their integration with other sources and its limitations,” *Renew. Energy*, vol. 230, no. 12, p. 120835, Sep. 2024, doi: 10.1016/j.renene.2024.120835.

[10] M. Mamuda and S. Sathasivam, “Comparing Logic Programming in Radial Basis Function Neural Network (RBFNN) and Hopfield Neural Network,” *International Journal of Computational and Electronic Aspects in Engineering*, vol. 1, no. 1, Dec. 2014, doi: 10.26706/ijceae.1.1.20141204.

[11] R. R. Abo-Alsabeh and A. Salhi, “The Genetic Algorithm: A study survey,” *Iraqi Journal of Science*, vol. 63, no. 3, pp. 1215–1231, 2022, doi: 10.24996/ijs.2022.63.3.27.

[12] M. Omari, M. Kaddi, K. Salameh, and A. Alnoman, “Advancing Image Compression Through Clustering Techniques: A Comprehensive Analysis,” *Technologies 2025, Vol. 13,* vol. 13, no. 3, Mar. 2025, doi: 10.3390/technologies13030123.

[13] “System-Level Power Optimization for Wireless Multimedia Communication,” *System-Level Power Optimization for Wireless Multimedia Communication*, 2002, doi: 10.1007/b117504.

[14] L. J. Saiz-Adalid, J. Gracia-Morán, D. Gil-Tomás, J. C. Baraza-Calvo, and P. J. Gil-Vicente, “Reducing the Overhead of BCH Codes: New Double Error Correction Codes,” *Electronics 2020, Vol. 9,* vol. 9, no. 11, pp. 1–15, Nov. 2020, doi: 10.3390/electronics9111897.

[15] S. I. Ahmed, “Performance Enhancement of 5G and 6G Mobile Communication Networks with Cooperative Non-Orthogonal Multiple Access and Caching.”

[16] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, “LSB: A Lightweight Scalable BlockChain for IoT Security and Privacy,” *ArXiv e-prints*, pp. 1–17, 2017, Accessed: Feb. 24, 2026. [Online]. Available: http://arxiv.org/abs/1712.02969

[17] L. Kong, H. Liu, Y. Shi, and J. Miao, “An Efficient Two-Stage Decoding Scheme for LDPC-CRC Concatenated Codes,” *Entropy 2025, Vol. 27,* vol. 27, no. 9, Aug. 2025, doi: 10.3390/e27090899.

[18] S. Jasim and A. Abbas, “Performance of Turbo Code with Different Parameters,” 2017.

[19] A. L. Bulgakov, A. V. Aleshina, S. D. Smirnov, A. D. Demidov, M. A. Milyutin, and Y. Xin, “Scalability and Security in Blockchain Networks: Evaluation of Sharding Algorithms and Prospects for Decentralized Data Storage,” *Mathematics 2024, Vol. 12,* vol. 12, no. 23, Dec. 2024, doi: 10.3390/math12233860.

[20] K. Hsieh, Y. W. Lin, S. I. Chu, H. C. Chang, and M. Y. Cho, “A Simple Neural-Network-Based Decoder for Short Binary Linear Block Codes,” *Applied Sciences 2023, Vol. 13,* vol. 13, no. 7, Mar. 2023, doi: 10.3390/app13074371.

[21] D. V. Ganin, M. A. Y. Damdam, and A. L. Savkin, “Permutation Decoding in Low-Power Wireless Sensor Networks,” *Автоматизация процессов управления*, vol. 2, no. 68, pp. 37–44, 2022, doi: 10.35752/1991-2927-2022-2-68-37-44.

[22] C. Liu, Y. Li, and T. Xu, “A Neural Network Approach to Physical Information Embedding for Optimal Power Flow,” *Sustainability 2024, Vol. 16,* vol. 16, no. 17, Aug. 2024, doi: 10.3390/su16177498.