Original scientific paper
https://doi.org/10.33180/InfMIDEM2025.206

Informacije MIDEM
Journal of Microelectronics, Electronic Components and Materials
Vol. 55, No. 2(2025), XX – XX

An efficient Spotted Hyena Optimizer based Multi-user Detection for Polar Encoder

Biju George George Alexander [1], Ablin Rino [2], Albert Raj Anthonymuthu [3]

[1]Assistant Professor, Department of Electronics and Communication Engineering, Stella Marys College of Engineering, Aruthenganvilai, Tamil Nadu, India
[2]Associate Professor, Department of Electronics and Communication Engineering, Arunachala College of Engineering for Women, Manavilai, Tamil Nadu, India
[3] Principal, Department of Electronics and Communication Engineering, DMI College of Engineering, Aralvaimozhi, Tamil Nadu, India

Abstract: Polar codes are among the most efficient types of error correction coding. Currently, these codes are employed in 5G communication networks and are the leading contender for 6G. Symmetry is significant in coding and decoding techniques for polar codes. However, some algorithms have high latency, and low throughput but suffer from high computational complexity. To overcome these issue a novel efficient Spotted Hyena Optimizer based Multi-User Detection for Polar Encoder (SHO-MUD) has been proposed for enhancing the throughput and reduce the latency. To increase spectrum, throughput, and energy efficiency, the SHO-MUD technique that has been suggested combines a polar encoder (PE) multiplexed with OFDMA with parity check polar coding (PCPC). PCPC-PE uses a system-configurable transmission rate to increase diversity gain and coding process dependability. To achieve optimum resource use over several data blocks, users are scheduled using the Spotted Hyena Optimizer (SHO) approach in conjunction with the MPA. The SHO scheduling efficiently allocates and schedules resources, resulting in a throughput gain of 0.6 bits per second. The suggested system provides user fairness by assuring an equal throughput of 1.55 bits/sec for all users.

Keywords: Polar Encoder; Multi-User Detection; Message Passing Algorithm: Spotted Hyena Optimizer; parity check polar coding

Učinkovito zaznavanje več uporabnikov za polarni kodirnik na osnovi Spotted Hyena optimizatorja

Izvleček: Polarne kode so ena najučinkovitejših vrst kodiranja za popravljanje napak. Trenutno se te kode uporabljajo v komunikacijskih omrežjih 5G in so glavni kandidat za 6G. Pri tehnikah kodiranja in dekodiranja polarnih kod je pomembna simetrija. Nekateri algoritmi imajo visoko latenco in nizko prepustnost, vendar trpijo zaradi visoke računske zapletenosti. Za odpravo teh težav je bil predlagan nov učinkovit optimizator Spotted Hyena za zaznavanje več uporabnikov (SHO-MUD) za povečanje prepustnosti in zmanjšanje zakasnitve. Za povečanje spektra, prepustnosti in energetske učinkovitosti predlagana tehnika SHO-MUD združuje polarni kodirnik (PE), ki je multipleksno povezan z OFDMA s polarnim kodiranjem za preverjanje paritete (PCPC). PCPC-PE uporablja sistemsko nastavljivo prenosno hitrost za povečanje raznolikosti in zanesljivosti procesa kodiranja. Za doseganje optimalne uporabe virov v več podatkovnih blokih se uporabniki načrtujejo z uporabo pristopa Spotted Hyena Optimizer (SHO) v povezavi z MPA. Načrtovanje SHO učinkovito razporeja in načrtuje vire, kar omogoča povečanje prepustnosti za 0,6 bita na sekundo. Predlagani sistem zagotavlja pravičnost uporabnikov z zagotavljanjem enake prepustnosti 1,55 bitov/s za vse uporabnike.

Ključne besede: Polarni kodirnik; zaznava več uporabnikov; algoritem za posredovanje sporočil: Spotted Hyena; polarno kodiranje s preverjanjem paritete

** Corresponding Author's e-mail: Bijugeorge.prf@outlook.com*

1 Introduction

Polar codes (PC) have garnered significant interest for their capacity to accomplish successive cancellation (SC) decoding and symmetric capacity on binary-input discrete memoryless channels. [1, 2]. After properly combining the input bits of several copies of the channel in a parallel concatenation scheme, the bit-channels present different reliabilities [3]. Asymptotically, as the number of copies goes to infinity, a fraction of the

bit-channels becomes noiseless, while the remaining bit channels are rendered useless [4,5]. Encoding amounts to sending the data bits through the good bit-channels and the so-called frozen bits (usually zeros) through the bad bit-channels [6,7].

Polar codes are intended to separate bit channels into the most and least reliable categories [8]. The least trustworthy channels are considered frozen, meaning they record zero values while creating code words. The most reliable channels are utilized to move information bits [9]. The way the polar codes are constructed determines how reliable the bit sequence is. There exist methods for generating convolutional polar codes [11], even if the majority of studies employ the traditional block structure given in [10]. Furthermore, polar codes are joined with other codes to create a concatenated code, which boosts productivity. A cyclic redundancy check (CRC) is therefore computed before to polar encoding in 5G NR, which significantly enhances transmission noise immunity. [12,13].

In 5G communication model, PC play a crucial role in several aspects, primarily in ensuring reliable and efficient data transmission over wireless channels. PC are utilized as the channel coding scheme in the control channels of the 5G New Radio (NR) standard [14]. These control channels are responsible for transmitting critical signaling information such as synchronization signals, system information, and scheduling assignments [15]. By employing polar codes, 5G systems can achieve high reliability and efficiency in transmitting these control signals, even in challenging wireless environments.

The reliability order of bit-channels is not universal since it is determined by channel circumstances and code length. Several approaches for designing frozen sets on the fly with little complexity have been offered. The major contribution of the work has been followed by

- The proposed approach combines parity check polar coding (PCPC) with a polar encoder (PE) and OFDMA multiplexing to improve spectral, throughput, and energy efficiency.
- PCPC-PE to enhance the coding process reliability and diversity gain by employing a flexible transmission rate on the system.
- To ensure proper resource utilization of multiple data blocks, the users are scheduled by implementing the Spotted Hyena Optimizer (SHO) technique along with the MPA.

The remaining portion of the work has been followed by, section 1 illustrates the introduction, section 2 represents the literature review, Section 3 illustrates the proposed model, Section 4 describes the findings from the experiment, and Section 5 depicts the conclusion of the work.

2 Literature Review

In 2021 Krasser, F.G., et al., [16] provided a unique and effective test bench and implementation for a small PC using an Intel DE10-Standard Development Kit for a System on Chip Field Programmable Gate Array (SoC FPGA). An 11% boost in throughput over a reference implementation for short PC is achieved by adopting fully-unrolled encoder and decoder architectures, which also result in high throughputs while consuming very little energy.

In 2023 Zhai, Y., et al., [17] Developed a polar code generation approach for the underwater acoustic channel that significantly decreases complexity while meeting the necessity for practical UWC. The suggested construction method and scheme also effectively ensures data transmission reliability, as demonstrated by lake-trial results under two different channel conditions.

In 2024 Pillet, C., et al., [18] describe a low-latency decoding strategy for shorter PC based on automorphism groups. The automorphism group of shorter polar codes, generated by two known shortening patterns, is shown to be finite but not empty, allowing for reduced polar code decoding with the Automorphism Ensemble (AE). Extensive simulation results for shorter polar codes using AE are shown and compared to the SC-List (SCL) approach. Shorter polar codes under AE have a block-error rate that matches or surpasses SCL while lowering decoding latency.

In 2023 Shreshtha, A. and Sarangi, S.R., [19] recommends an innovative way for appropriately using PC encoders in a 5G base station. Additionally, we offer a collection of novel resource allocation algorithms and assess their efficacy against similar methods in the literature in order to intelligently allocate user data packets across available compute nodes. Using our suggested optimization approaches, we save 17% on a 5G base station. Simultaneously, we may enhance performance by 24% over a typical base station.

In 2019 Sharma, A. and Salim, M., [20] focused on channel coding schemes, specifically for URLLC use cases in 5G New Radio (5G-NR), and analyzing the performance of polar codes for the same. Polar codes are examined using a variety of performance settings for short block lengths and low code rates, as required for the URLLC scenario. The extremely reliable polar code's exceptional error correction performance, along with its low computational complexity and decoding delay, makes it a viable candidate in the URLLC channel coding competition.

In 2021 Liao, Y., et al., [21] proposed a novel polar-code generation approach that avoids the need to sift and select bit channels depending on dependability. It is proved that using this strategy to optimize polar-code synthesis for the SCL decoder is equivalent to maximizing the anticipated benefit of traversing a maze. The simulation findings reveal that typical polar-code architectures built for successive cancellation decoders are no longer ideal for SCL decoding in terms of frame error rate.

In 2020 Xu, W., et al., [22] suggested deep learning (DL) techniques to optimising polar belief propagation (BP) decoding and concatenated LDPC-polar codes. Numerical simulations reveal that there is no performance difference between 2-D OMS and accurate BP at varied code lengths.

In 2022 Tseng, S.M., [23] employing LSTM for Polar code decoding with Markov Gaussian memory impulse noise channels. The proposed LSTM-based approach has a one-third bit error probability when compared to typical SC/BP/SCL decoding algorithms for Markov Gaussian channels. It is also 5–12 times faster in execution time and decoding delay.

In 2019 Wen, C., et al., [24] presented a technique that combines convolutional neural networks (CNN) and classic Belief Propagation (BP) decoding. We model the proposed method's performance using 4QAM and BPSK modulation. The traditional real-valued CNN approach is extended to complex-valued ones using a QAM modulation method that is detailed. The findings indicate that lowering anticipated noise increases bit error rate (BER) performance.

In 2023 Hebbar, S.A., et al., [25] provide a novel CurRIculum-based Sequential neural decoder for Polar coding (CRISP). The recommended curriculum is critical in increasing CRISP accuracy, as seen by comparisons to other curricula. To the best of our knowledge, CRISP is the first data-driven decoder for PAC codes, and it works almost optimally on the PAC (32,16) code.

3 Proposed Methodology

In this paper a novel efficient Spotted Hyena Optimizer based Multi-User Detection for Polar Encoder (SHO-MUD) has been proposed for enhancing the throughput and reduce the latency

Polar codes

Erdal Arıkan (Arikan, 2009) developed polar codes, the first deterministic code architecture that achieves Shannon capacity with minimal encoding and decoding complexity. This section introduces PC and gives inspiration for our technique.

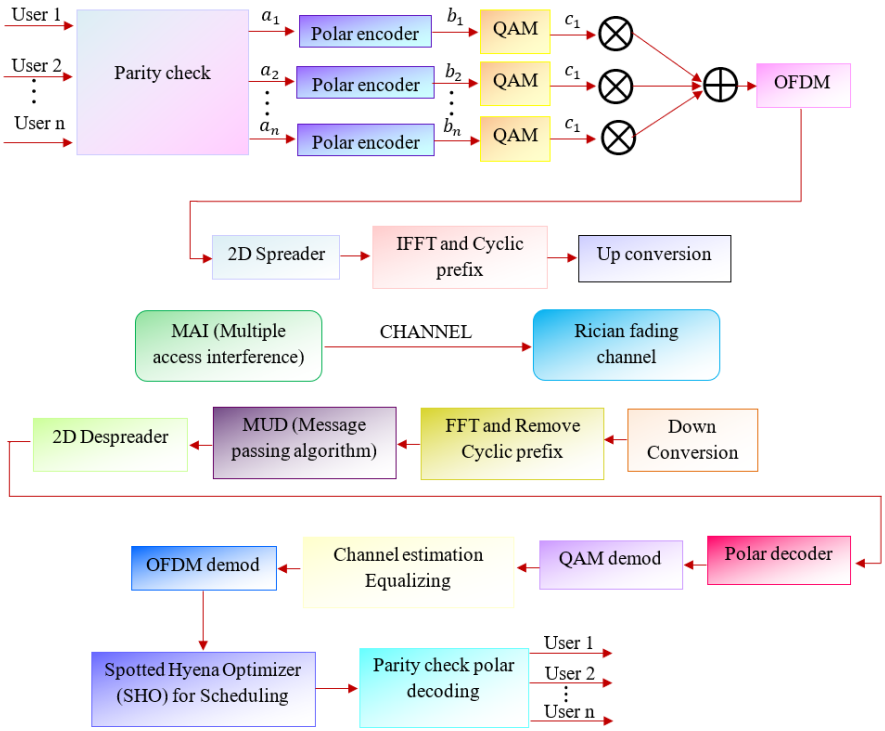


Figure 1: Proposed Methodology

3.1 Polar Encoding

One linear block coding scheme, PC, encodes by modifying linear codes into systematic codes and decodes in a range of scenarios with little to no complexity. The use of a PCPC determines the system's dependability utilized a Polarization Weight (PW) evaluation and enhances BER performance. The following equations describe the coding system for linear transformations over the field F.

$$U_0^{a-1} = b_o^{k-1} G_1 \quad (1)$$

Where U_0^{a-1} represents the code word, b_o^{k-1} represents the message word, and G_1 represents the generator matrix using eqn 1

$$b_o^{k-1} = b_o^{k-1} Y, b_o^{k-1} Y_u \quad (2)$$

The first part of $b_o^{k-1} Y$ includes user information provided by to enable a free interchange between each transmission cycle.

$$b_o^{k-1}Y = b_o^{k-1_j} : j \in Y \quad (3)$$

The number of digits that are considered frozen at the start of the decoding process is provided in the second section and is shown by

$$b_o^{k-1}Y_u = b_o^{k-1_j} : j \in Y_u \quad (4)$$

Combine the equation 3,4, and 5 will written by

$$U_0^{a-1} = b_o^{k-1}YG_{1Y} + b_o^{k-1}Y_uG_{1Y_u} \quad (5)$$

where, G_{1Y} and G_{1Y_b} illustrates the sub-matrices of $G_1$with row indices Y and Y_u . The non-systematic encoder is the name given to this kind of mapping. By appropriately selecting the set Y's size, the coding rate may be adjusted.

$$U_0^{a-1} = b_o^{k-1}YG_{1YX} + b_o^{k-1}Y_uG_{1Y_{c'}} \quad (6)$$

$$U_0^{a-1} = b_o^{k-1}YG_{1YX_u} + b_o^{k-1}Y_uG_{1Y_uX_{u'}} \quad (7)$$

$$M = 2^m \quad (8)$$

The generator matrices PC with block size m is given as

$$G = H^{\otimes m} \quad (9)$$

where, $H = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ illustrates the Kernel, and $\otimes$ denotes the Kronecker power.

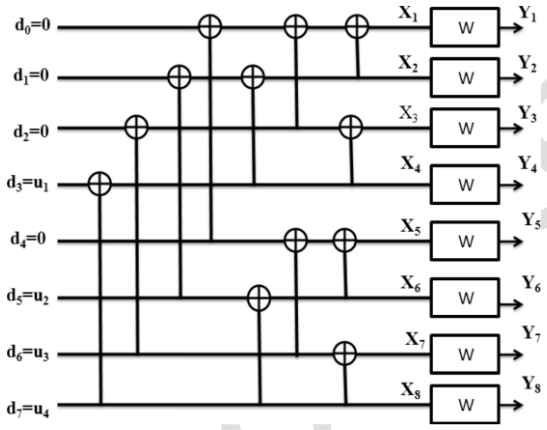


Figure 2: Polar codes encoder

Figure 2. depicts a model for PC encoding operation with (N, K, C^C) and encoding decoding complexity. $O(NlogN)$.

3.1.1 2D Spreader

The 2D spreader receives the output from the SCMA encoder after that. Two-dimensional OFDM distribution allows for frequency reuse and frequency variation in a cellular environment. By allocating a portion of subcarriers to every receiver, OFCDM may provide multiple-access. Additionally, there is an increase in throughput along with temporal and frequency diversity. The provided symbol may be distributed using codes {+1, −1, +1, −1} in the time domain and {+1, −1} in the frequency domain. Effective transmission across a multipath fading channel is made possible by the 2D spreader.

3.1.2 IFFT and Cyclic prefix

Symbols that are sent are directed to the IFFT block, which performs 2D spreading and converts the frequency domain vector signal into a time domain signals. Guard intervals help prevent fading from many paths. The 2D spread data streams are fed into the IFFT block in the following manner to provide the orthogonality of subcarriers:

$$k(t) = \sum_{c=-\infty}^{\infty} \sum_{v=0}^{V-1} l_{c,v}\, Rect(t - mT) e^{j2\pi c/T} \quad (10)$$

where $l_{c,v}$ represents the cth transmitted symbol in the vth subcarrier. Prior to transmission to the base station, the signal is up-converted.

3.1.3 Multi-User Detection (MUD) via Message passing Algorithm

MUDs are used on the receiving end to separate user data from non-orthogonal overlapping data. The number of data layer collisions on each resource piece is decreased by the sparseness of the polar codewords. Multiple Access Interference (MAI) is the main factor affecting MUD performance. The best performing but most difficult approach for handling MAI is the optimal maximal a posteriori algorithm (MAP). A straightforward technique is needed to improve MAP performance. The serial MPA is an efficient method for scheduling in ascending order on the receiving side. The serial MPA algorithm's convergence rate varies with scheduling, though. Consequently, an IWO-based MPA detection system is presented to improve its performance, which creates a path between the resource node and user node for an appropriate scheduling order.

3.2 Polar Decoding

A technique for deciphering communications in polar codes is called polar decoding. PC are a kind of error-correcting codes that express data using polar coordinates. Polar decoding involves converting the received signal's polar coordinates to Cartesian coordinates before applying a typical decoding method to decode it. Applications for polar decoding are numerous and include data storage and wireless communication.

3.2.1 Rician Fading Channel

The Rician fading version is another statistical variant that implies that the sign consists of a resilient LOS element and a random component. The LOS element is a linear channel with regular amplitude and segment that connects the transmitter and receiver.

$$P = \frac{u^2}{2\sigma^2} \quad (11)$$

$P = \frac{u^2}{2\sigma^2}$described as the scale parameter andRatio of power contributions from line-of-sight paths to other multipaths.

The second scaling factor for the distribution is Ω, which represents the total power from both pathways.

$$\Omega = u^2 + 2\sigma^2 \quad (12)$$

$\Omega = u^2 + 2\sigma^2$described as the total power obtained across all routes. The received signal amplitude (rather than the received signal strength) is then distributed using the following parameters in rice distribution (RD). The function of probability density is

$$u^2 = \frac{P}{1+P}\Omega \quad (13)$$

$$\sigma^2 = \frac{\Omega}{2(1+P)} \quad (14)$$

$$f(a|u,\sigma) = \frac{a}{\sigma^2} exp\left(\frac{-(a^2+u^2)}{2\sigma^2}\right) I_0\left(\frac{au}{\sigma^2}\right) \quad (15)$$

This leads to the following probability density function:

$$f(a) = \frac{2(P+1)a}{\Omega} \exp\left(-P - \frac{(P+1)a^2}{\Omega}\right) I_0\left(2\sqrt{\frac{P(P+1)}{\Omega}}a\right) \quad (16)$$

In this case, I_0 is the first-kind modified Bessel function of order zero at 0th order.

3.3 Spotted Hyena Optimizer (SHO)

The behavior of the spotted hyena, particularly its social relationships with other animals, served as inspiration for the SHO. The spotted, striped, brown, and aardwolf are the four recognized hyena species; they differ in size, behavior, and nutrition. Out of the three hyena species, the spotted hyena is the most proficient predator. Spotted hyena females live in clans or groups.

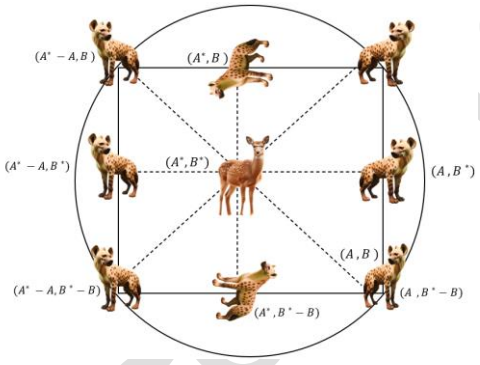


Figure 3: Position vectors in two dimensions of spotted hyena

To create SHO, the hunting tactics and social dynamics of the spotted hyena are computationally constructed. The four stages of SHO include encircling, hunting, attacking prey, and searching. Spotted hyena's position vectors in 2d are given in figure 3.

3.3.1 Encircling

In SHO, hyenas try to place themselves as close to the prey as possible, leading the group to that location. The optimal position is now approximated to be the location of the prey. Once they locate their meal, hyenas circle around it. First, the best member of the population is recognized, and others modify their views accordingly. The encircling mechanism is modeled by equation (17).

$$\overrightarrow{D\iota_{hy}} = \left|\vec{A} \cdot \overrightarrow{Po_{py}}(y) - \overrightarrow{Po}(y)\right| \quad (17)$$

$$\overrightarrow{Po}(y+1) = \overrightarrow{Po_{py}}(y) - \vec{F} \cdot \overrightarrow{D\iota_{hy}} \quad (18)$$

Equation (2) defines parameter Dt as the distance between a hyena and the prey's position. $\overrightarrow{Po}(y+1)$ denotes the hyena's newfangled location in the current repetition. The current iteration is denoted by y in Equations (18) and (19), The algorithm iteration is y, and hyenas use this equation to gauge the distance to their prey, surrounded by specific mechanisms. Equations (20) and (21) calculate vector coefficients $\vec{A}$ and $\vec{F}$, using position vectors and element-wise multiplication.

$$\vec{A} = 2 \cdot \overrightarrow{qb_1} \quad (19)$$

$$\vec{F} = 2\vec{d} \cdot \overrightarrow{qb_2} - \vec{d} \quad (20)$$

$$\vec{d} = 5 - \left(IT \times \left(\frac{5}{max_{IT}}\right)\right) \quad (21)$$

Equation (5) shows that $\overrightarrow{qb_1}$ and $\overrightarrow{qb_2}$ are random vectors in the interval (0, 1), and that h drops linearly from 5 to 0 during the iteration time. The maximum number of iterations is indicated by the max_{IT} option. An increased number of iterations is necessary to increase efficiency.

3.3.2 Hunting

Hyenas are mostly gregarious animals that hunt in packs and are adept at locating prey. In order to statistically explain hyena behavior, it is thought that the optimal search factor is an ideal component that knows the position of the prey. In order to identify the top search agent, other search agents' band together and update the best results they have found which are given in equation (22), (23) and (24).

$$\overrightarrow{D\iota_{hy}} = \left|\vec{A} \cdot \overrightarrow{Po_{hy}} - \overrightarrow{Po_k}\right| \quad (22)$$

$$\overrightarrow{Po_k} = \overrightarrow{Po_{hy}} - \vec{F}\,\overrightarrow{D\iota_{hy}} \quad (23)$$

$$\overrightarrow{C_{hy}} = \overrightarrow{Po_k} + \overrightarrow{Po_{k+1}} + \cdots + \overrightarrow{Po_{k+M}} \quad (24)$$

The first hyena's ideal location is determined by the constraint $\overrightarrow{Po_{hy}}$, whereas the other hyenas' location is indicated by $\overrightarrow{Po_k}$. Equation (10) is used to calculate the number of hyenas, which is displayed in Parameter M.

$$M = count_{ns}\left(\overrightarrow{Po_{hy}} + \overrightarrow{Po_{hy+1}} + \overrightarrow{Po_{hy+2}}, \ldots\ldots, \left(\overrightarrow{Po_{hy}} + \vec{N}\right)\right) \quad (25)$$

The ns parameter in Equation (25) specifies the total number of solutions as well as all feasible solutions., where $\vec{N}$ is a random vector in the interval (0.5,1). The parameter $\overrightarrow{C_{hy}}$ represents a set of the ideal solution's number M.

3.3.3 Attacking Prey

The worth of the path h_y is decreased with the purpose of creating a mathematical model for the target attack. The vector h can have its value reduced from 5 to 0 during an iteration by reducing the change in the vector F. The pack of untainted hyenas is forced to outbreak the target if the value of F is |F| < 1, Which is given in equation (26).

$$\overrightarrow{Po}(y+1) = \frac{\overrightarrow{C_{hy}}}{M} \quad (26)$$

The ideal position is saved and updated by $\overrightarrow{Po}(y+1)$ in Equation (26), and The best agent's position determines how the other search agents are positioned.

3.3.4 Searching for prey

This method depends on modifications to the vector $\vec{F}$ that enable hunting (random search). The arbitrary numbers that force the search engine to depart from a reference hyena and are greater than or less than −1 are represented by $\vec{F}$. When compared to the non-random search phase, the random search phase computes a search agent's location based on the search agent's selection at random rather than using the current best search agent. The SHO may conduct a global search thanks to this mechanism and |F| > 1. Both of these factors highlight random search.

4 Materials and Methods

In this section evaluates and compares the performance of the proposed SHO-MUD. The experimental arrangement executed on PC with Windows 11, 13 generation Intel (R) Core (TM) is 1335U 1.30 GH2.

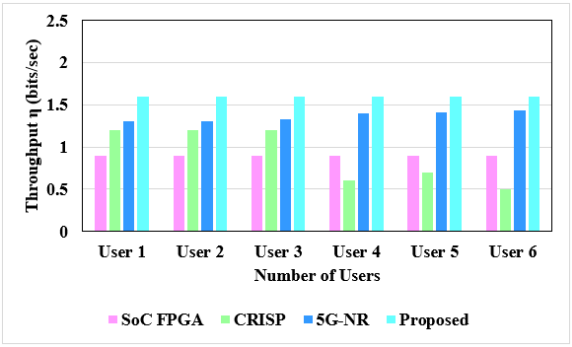


Figure 4: Number of user vs Throughput

Figure 4 depicts the analysis of throughput for each user. In this scenario, the suggested throughput results are compared to benchmark methodologies such as SoC FPGA [16], CRISP [25], and 5G-NR [20]. The SHO scheduling efficiently allocates and schedules resources, resulting in a throughput gain of 0.6 bits per second. The suggested system provides user fairness by assuring an equal throughput of 1.55 bits/sec for all users. The overall Energy Efficiency of the proposed SHO-MUD method is 1.6%. The throughput of the proposed model 1.34%, 1.36%, and 0.9% better than SoC FPGA, CRISP and 5G-NR respectively.

Figure 5 shows the variance in outage probability over the Rician fading distribution. The projected outage probability is calculated depending on the number of interfering signals. The proposed method is 19% whereas, existing SoC FPGA has 18.6%, CRISP has 48.7% and 5G-NR has 51.23% respectively.

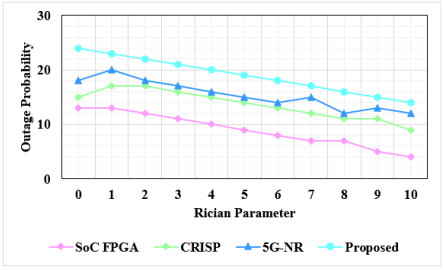


Figure 5: Rician Fading Distribution Outage Probability

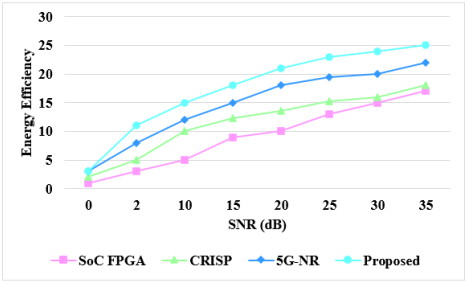
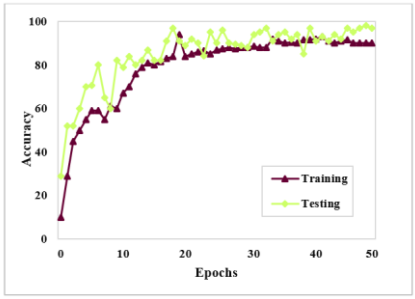


Figure 6: Energy Efficiency

Fig. 6, all techniques show an increase in energy efficiency as SNR increases. The suggested technique outperforms the existing SoC FPGA [16], CRISP [25], and 5G-NR [20] approaches in terms of EE. The average Energy Efficiency of the proposed SHO-MUD method is 31.1%. The BER of the proposed model 0.52%, 0.58%, and 0.89% better than SoC FPGA, CRISP and 5G-NR respectively.



(a)

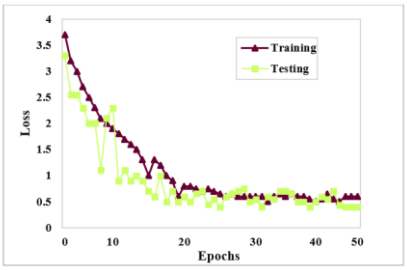


Figure 7: Performance based on training and testing sets

Figure 7(a,b) shows the horizontal axis as the number of times epoch. One epoch, out of all the examples, had one forward pass and one retrograde pass. The variation in accuracy and loss with increasing epochs is seen in Figure 7(a,b).

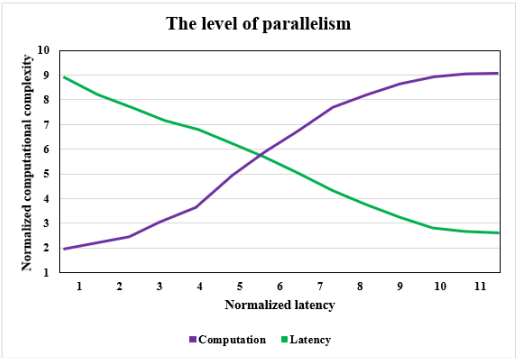


Figure 8: Normalized latency and computational complexity

Figure 8 displays the computational complexity and latency values of the suggested design with N = 32, 768, which are correspondingly normalized by the recommended values. When the decoding converges and the amount of parallelism grows, each point displays a normalized value. In comparison to the suggested decoding, the proposed decoding flexibility reduces delay at the expense of computational complexity.

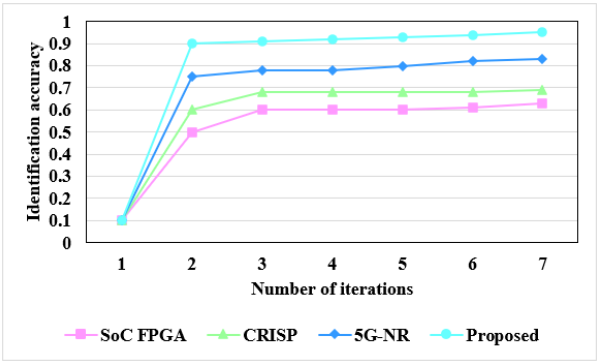


Figure 9: Convergence of iterations vs identification accuracy

Figure 9 illustrates the convergence of the pro-posed and the basic technique. It is evident from the graphic that the suggested strategy outperforms previous approaches in terms of identification accuracy. With a 90.71% accuracy the proposed method performs better the existing techniques such as SoC FPGA has 5.02%, CRISP has 4.96% and 5G-NR has 3.44% respectively.

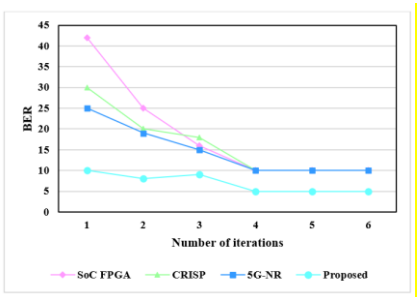


Figure 10: Number of iterations vs BER

Figure 10 displays a comparison of the detection scheme's convergence behavior. When compared to the current serial and parallel MPA approaches, the suggested scheme exhibits a greater rate of convergence after at least two rounds. The average BER of the proposed method is 7%. The BER of the proposed model 5.02%, 4.96%, and 3.44% lower than SoC FPGA, CRISP and 5G-NR respectively.

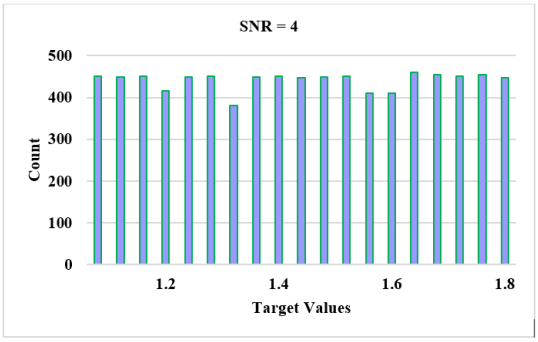


Figure 11: Target distribution for SNR = 4dB

Figure 12: Target distribution for SNR= -8 dB

Fig. 11 displays the values' histogram at two distinct SNR levels, namely SNR=4 dB The target distribution for SNR = -8 dB is displayed in Figure 12. Every SNR received its own file, which was made. As a result, building models for various SNR levels is made simpler. Furthermore, a larger random collection of SNR values has been generated, which may be utilized to train a global model that can forecast for any SNR value.

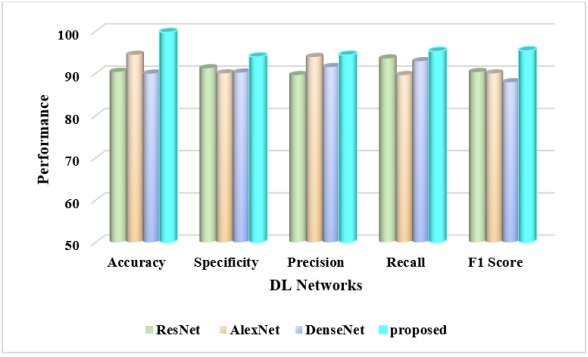


Figure 13: Comparison analysis of existing deep learning models

The accuracy obtained by ResNet, Alex Net, DenseNet, and suggested is 90.34%, 94.39%, 89.92%, and 99.5%, respectively, as Figure 13 illustrates. The suggested specificities of 91.19%, 89.98%, 90.18%, and 99.7% were attained by ResNet, Alex Net, and DenseNet. ResNet, Alex Net, and DenseNet yielded the following recommended precision values: 89.58%, 93.87%, 91.49%, and 94.29%. ResNet, Alex Net, and DenseNet yielded the following projected recall percentages: 93.51%, 89.59%, 92.87%, and 95.18%. ResNet, Alex Net, and DenseNet provide the F1 score, which is 90.31%, 89.98%, 87.89%, and 95.34%, respectively. The accuracy rate of the suggested is greater than that of the models in use at the moment.

5 Conclusions

In this section a novel efficient Spotted Hyena Optimizer based Multi-User Detection for Polar Encoder (SHO-MUD) has been proposed for enhancing the throughput and reduce the latency. Using a SHO-based MUD method, the suggested technique lowers complexity and increases convergence rate in terms of iterations. It also lowers the Bit Error Rate (BER) and achieves improved throughput and energy efficiency, respectively. The SHO scheduling effectively schedules and distributes the resources, increasing throughput by 0.6 bits/sec. By guaranteeing an equal throughput of 1.55 bits/sec for every user, the suggested technique accomplishes user fairness. The EE of the proposed method is17%, 18%, 23.2%, and 23% better than existing techniques. Future research aims to assess how well the suggested generalized rapid decoding applies to real-world multi-kernel codes.

6 Acknowledgments

The author would like to express his heartfelt gratitude to the supervisor for his guidance and unwavering support during this research for his guidance and support.

7 Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

8 References

1. J. Wang, and C. Ling, "Polar sampler: A novel Bernoulli sampler using polar codes with application to integer Gaussian sampling," *Designs, Codes Cryptogr.* vol. 91, no. 5, pp. 1779-1811, 2023. http://dx.doi.org/10.1007/s10623-022-01164-7
2. K. B. Priya Iyer, S. Ramesh, J. Sathiamoorthy, & A. Ahilan, "Tetra Optimization Based Hybrid Parameters for OFDM Modulated Wireless Sensor Network", *IETE J. Res.*, vol. 70, no. 5, pp. 4509–4522, 2023, https://doi.org/10.1080/03772063.2023.2225471
3. C.J. Pang, "Resource-constrained Coding for Communication and Computation Applications (Doctoral dissertation)". 2023. http://dx.doi.org/10.1007/978-3-642-20520-0_6
4. M. Prabhu, B. Muthu Kumar & A. Ahilan, Slime Mould Algorithm based Fuzzy Linear CFO Estimation in Wireless Sensor Networks. IETE J. Res., 1-11 (2023).https://doi.org/10.1080/03772063.2023.2194279
5. K. Niu, P. Zhang, J. Dai, Z. Si, and C. Dong, "A golden decade of polar codes: From basic principle to 5G applications," *China Commun.* vol. 20, no. 2, pp. 94-121, 2023. http://dx.doi.org/10.23919/jcc.2023.02.015
6. Y. Deng, S. Wu, J. You, J. Jiao, N. Zhang, and Q. Zhang, "Optimizing age of information in Polar coded status update system," *IEEE Internet Things J.* 2023. http://dx.doi.org/10.1109/jiot.2023.3290644
7. H. Hematkhah, Y. Kavian, and E. Namjoo, "PoCH: automatic HDL code generator tool for Polar channel coding decoders in multimedia communication systems," *Multimedia Tools Appl.* vol. 82, no. 24, pp. 36739-36768, 2023. http://dx.doi.org/10.1007/s11042-023-14507-w
8. M. Fletcher, E. Paulz, D. Ridge, and A.J. Michaels, "Low-Latency Wireless Network Extension for Industrial Internet of Things," *Sens.* vol. 24, no. 7, pp. 2113, 2024. http://dx.doi.org/10.3390/s24072113
9. A. Alashqar, J. Alkasassbeh, R. Mesleh, and A. Al-Qaisi, "SDR implementation and real-time performance evaluation of 5G channel coding techniques," *AEU Int. J. Electron. Commun.* vol.

170, pp. 154852, 2023. http://dx.doi.org/10.1016/j.aeue.2023.154852

10. K. Pelekanakis, P. Paglierani, A. Alvarez, and J. Alves, "Comparison of error correction codes via optimal channel replay of high north underwater acoustic channels," *Comput. Network.* 110270, 2024. http://dx.doi.org/10.1016/j.comnet.2024.110270
11. D. Yang, Y. Mao, and X. Liu, "Judgement of error frames using frozen bits and its applications in decoding of polar codes," *IET Commun.* vol. 17, no. 14, pp. 1760-1772, 2023. http://dx.doi.org/10.1049/cmu2.12651
12. T. Jiang, Y. Liu, L. Xiao, W. Liu, and G. Liu, "PCC polar codes for future wireless communications: Potential applications and design guidelines," *IEEE Wireless Commun.* 2023.http://dx.doi.org/10.1109/mwc.017.2200586
13. S. Lydia, Y.K. Gultom, S. Alam, I. Surjati, A. Alkina, A. Mekhtiyev, Y. Neshina, T. Serikov, P. Madi, K. Sansyzbay, and A. Yurchenko, "Polar Code Performance Analysis for High-Speed Wireless Data Communication System," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 05, 2022.http://dx.doi.org/10.21883/tpl.2022.08.55116.19200
14. M. Maksimovi, and M. Forcan, "Application of 5G channel coding techniques in smart grid: LDPC vs. polar coding for command messaging," *In 7th International Conference on Electronics, Telecommunications, Computing, Automatics and Nuclear Engineering-IcETRAN,* pp. 8-10, 2020. http://dx.doi.org/10.1016/j.segan.2021.100495
15. N.A. Mohammed, A.M. Mansoor, R.B. Ahmad, and S.R.B. Azzuhri, "Deployment of polar codes for mission-critical machine-type communication over wireless networks," arXiv preprint arXiv:2110.02938. 2021.http://dx.doi.org/10.32604/cmc.2022.020462
16. F.G. Krasser, M.C. Liberatori, L. Coppolillo, L. Arnone, and J.C. Moreira, "Fast and efficient FPGA implementation of Polar Codes and SoC test bench," *Microprocess, Microsyst.* vol. 84, pp. 104264, 2021. http://dx.doi.org/10.1016/j.micpro.2021.104264
17. Y. Zhai, J. Li, H. Feng, and F. Hong, "Application research of polar coded OFDM underwater acoustic communications," *EURASIP J. Wireless Commun. Networking,* vol. 2023, no. 1, pp. 26, 2023. http://dx.doi.org/10.1186/s13638-023-02236-5
18. C. Pillet, I. Sagitov, V. Bioglio, and P. "Giard Shortened Polar Codes under Automorphism Ensemble Decoding," IEEE Commun. Lett. 2024. http://dx.doi.org/10.1109/lcomm.2024.3365091
19. A. Shreshtha, and S. R. Sarangi, "Efficiently Using Polar Codes in 5G Base Stations to Enhance Rural Connectivity." arXiv preprint arXiv:2306.15476. 2023. http://dx.doi.org/10.1063/pt.5.028530
20. A. Sharma, and M. Salim, "Polar code appropriateness for ultra-reliable and low-latency use cases of 5G systems," *Int. J. Networked Distrib. Comput.* vol. 7, no. 3, pp. 93-99, 2019. http://dx.doi.org/10.2991/ijndc.k.190702.005
21. Y. Liao, S.A. Hashemi, J.M. Cioffi, and A. Goldsmith, "Construction of polar codes with reinforcement learning," *IEEE Trans. Commun.* vol. 70, no. 1, pp. 185-198, 2021. http://dx.doi.org/10.1109/tcomm.2021.3120274
22. W. Xu, X. Ta, Y. Be'ery, Y.L. Ueng, Y. Huang, X. You, and C. Zhang, "Deep learning-aided belief propagation decoder for polar codes," *IEEE J. Emerging Sel. Top. Circuits Syst.* vol. 10, no. 2, pp. 189-203, 2020. http://dx.doi.org/10.1109/jetcas.2020.2995962
23. S.M. Tseng, W.C. Hsu, and F. Tseng, "Deep learning-based decoding for polar codes in Markov Gaussian memory impulse noise channels," *Wireless Pers. Commun.* vol. 122, no. 1, 737-753, 2022. http://dx.doi.org/10.1007/s11277-021-08923-0
24. C. Wen, J. Xiong, L. Gui, Z. Shi, and Y. Wang, "A novel decoding scheme for polar code using convolutional neural network," *In 2019 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB),* pp. 1-5, 2019. IEEE. http://dx.doi.org/10.1109/bmsb47279.2019.8971888
25. S. A. Hebbar, V.V. Nadkarni, A.V. Makkuva, S. Bhat, S. Oh, and P. Viswanath, "CRISP: Curriculum based Sequential neural decoders for Polar code family," *In International Conference on Machine Learning,* pp. 12823-12845, 2023. PMLR. http://dx.doi.org/10.1109/isit50566.2022.9834589

Arrived: 15.07.2024
Accepted: 07.01.2025