IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 3 Issue 12, December 2016

ISSN (Online) 2348 – 7968 | Impact Factor (2015) - 4.332

www.ijiset.com

Hybrid Meta Heuristic Search Algorithm for High Dimensional Data Transmission

Rutuja A. Deshmukh[1] and Dr. Ashish R. Panat[2]

[1] Electronics Department, Priyadarshini College of Engineering,
Nagpur, Maharashtra, India

[2] Electronics Department, Priyadarshini College of Engineering,
Nagpur, Maharashtra, India

Abstract

Interleaver plays key role in preserving the performance of turbo encoding systems. Under high dimensional data transmission, the interleaver design often goes complex. This paper presents a hybrid meta-heuristic search algorithm by combining renowned Genetic Algorithm (GA) and Group Search Optimizer (GSO) in the name of hybrid GSO (HGSO). The HGSO is emphasized to operate in high dimensional space so that the interleaver design is expected to be robust under high dimensional data transmission. The hybridization embodies the mutation operator of GA in the GSO scanning process. This improves the exploration process of GSO to enable faster convergence. Experiments are conducted at higher order data bits and the performance of HGSO is demonstrated. A statistical report is prepared from the observed results to illustrate the reliability of the outcome accomplished by HGSO over the other methods.

Keywords: *interleaver; optimization; design; bit error rate; frame error rate*

1. Introduction

The turbo codes were introduced by Berrou, Glavieux and Thitimajshima in 1993 [1] and they have become a hot topic soon after their introduction [12] [15]. Prior to the turbo codes, 3dB or more separated the spectral efficiency of real world channel encoding systems from the theoretical maximum described by Shannon theorem [1]. Turbo coding brought to the world of channel encoding one important principle: the feedback concept, exploited heavily in electronics, to be utilized in decoding of concatenated codes and it was indeed the iterative decoding (the actual turbo principle) that helped the turbo codes to achieve its impressive nearoptimum performance [1]. The original turbo coder [1], [2] was designed as a parallel concatenation of two circular recursive systematic convolutional codes. Generally, the performance of the turbo codes depends on two principal parameters; the first is the code spectrum, and the second is the decorrelation between the external information at the same number of iterations.

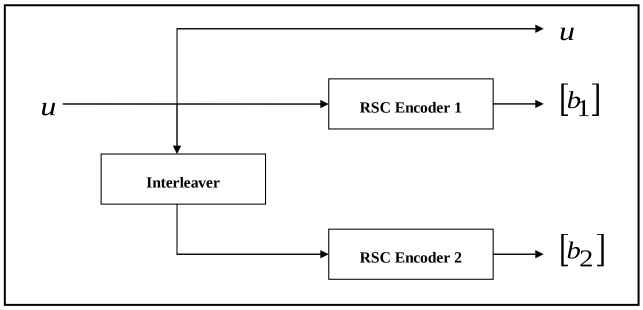


Fig. 1. Simple architecture of the turbo encoding scheme

The optimization process can be used for the amelioration of performance and the diminution of the matrix stature with safe performance. The latter is very interesting for multimedia real-time satellite transmission systems because the interleaving matrix causes a considerable diminution of the codec complexity and delay [4]. One of the key principles behind the performance of turbo coding is the fact that each of the encoders typically produces a high-weight code word for most inputs, but it produces a low-weight (i.e. more error prone) code word for only few inputs. The Turbo encoder is responsible for the generation of a high-performance convolutional code, the value of which moves very close to the channel capacity [17]. As depicted in fig. 1, it receives a serial input stream and sends three sub-blocks of bits, namely, payload data u, parity bits b_1, and b_2. The parity bits are computed by two Recursive Systematic Convolutional (RSC) encoders (RSC encoder 1 and RSC encoder 2), respectively. In order to generate different redundant sub-blocks of parity bits, an interleaver, as given in fig. 3, is used to force input bits to appear in different sequences. This action can deal with a burst of errors appearing in close proximity. The code word b is then combined by the three output sub-blocks one bit by one bit. Thus a code with redundant information is generated. In the simple RSC encoder illustrated in fig. 2, serial concatenated memory registers D are connected with XOR gates in such a way that the XOR gates are operated along with the feedback provided by the second register.

A. Interleaver on turbo encoder

The key component in turbo codes is the interleaver, which permutes the input frame before it is processed by the second encoder. The interleaver makes it more unlikely that both encoders will output a bad code word for the same input, increasing the probability that the decoder will be able to extract the correct information.

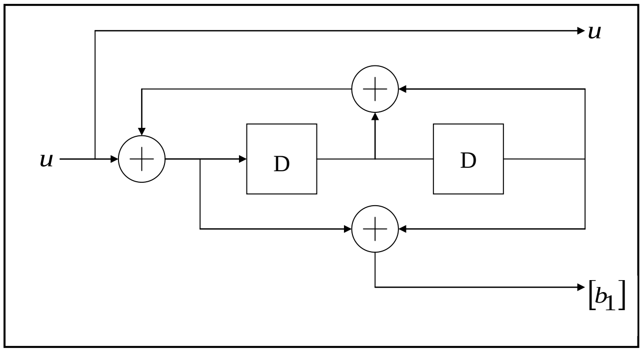


Fig. 2. RSC encoder 1

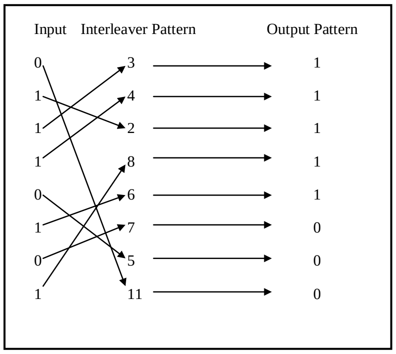


Fig. 3. Interleaving process

The effect of interleaver on turbo code performance was recognized in 1995 [7] and the actual importance of randomization was shown by Perez et al. [8] and Divsallar and Pollara [9], by considering the effect the interleaver on critical low weight input patterns.Divsallar and Pollara [5] also proposed the S-random interleaver, which is a randomly generated interleaver with some restrictions on the spread, designed to break low weight input patterns. A different interleaver design technique, the iterative decoding suitability (IDS) criterion was presented by Hokfelt et al. [10]. Their interleaver design is based on the statistical properties of the extrinsic information passed between the decoders rather than on the turbo code's distance properties. Sadjadpour et al. [10] has combined the two criteria to design a two step S-random interleaver. Recently some deterministic interleavers have been proposed [11]. These interleavers are designed to achieve maximum spread to correct single error events and certain amount of controlled disorder or vectorial fluctuation is provided to correct multiple error events. On the other hand, searching for an optimal interleaver for implementing with an interleaver matrix requires the dimension N to perform a hardware permutation of N input bits but it can be modeled as a general permutation of N symbols. Interleaver sizes vary from tens to tens of thousands. When, it is computationally infeasible to test all possible interleaver matrices (N!) and all possible input vectors (2N) to find a globally best interleaver [5]. Therefore, advanced interleaver optimization methods are sought. A search for an optimal permutation of N symbols is a typical combinatorial optimization problem. This paper attempts to introduce a robust interleaver design using hybrid meta-heuristic search algorithms. The hybridization is performed between GSO and GA and it is operated in high dimensional solution space so that high dimensional data transmission is ensured by the interleaver. The paper is organized as follows. Section II reviews the literature and states the problem, whereas Section III details the proposed interleaver design procedure. Section IV discusses the results and Section V concludes the paper.

2. Literature Review

Numerous methods on interleaving optimization have been reported in the literature.

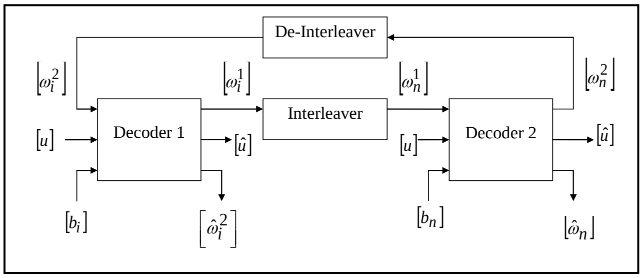


Fig. 4. Turbo decoder

Though all the contributions are not described here, few recent works on designing interleavers are presented here to demonstrate the significance. Laddomada, M.; Scanavino, B. [1] have worked on designing semi-random prunable interleavers for Parallel Concatenated Convolutional Codes (PCCC). In order to optimize the interleaver design, they have introduced an iterative algorithm. The iterative growth has started from a smaller interleaver till it reaches a desired length. Moreover, they have formulated a cost function by considering the correlation properties of the extrinsic information and the concept of spread of an interleaver. The cost function is tends to be minimized so that the interleaver optimization can be accomplished.

Muhammad Arif et al. [13] have introduced a circular shift in the permutation function to reduce the correlation between the parity bits corresponding to the original and interleaved data frames to improve the decoding capability of MAP decoder. The design was focused on combining good permutations with de-correlation property. Their solution to design a deterministic interleaver outperformed the semi-random interleavers and the deterministic interleavers reported in the literature.

Lucian Trifina & Daniela Tarniceriu [14] have proposed a method for searching interleavers within a certain class, with the aim of designing turbo codes with good distance spectrum. The method was based on a modified version of Garello's algorithm and consists in the calculation of frame error rate truncated upper bound.

Mohd Azri Mohd Izhar et al. [16] proposed a joint source-channel coding technique for two-dimensional (2D) binary Markov sources by using concatenated turbo block codes composed of two Bose, Chaudhuri, Hocquenghem (BCH) codes, of which output is followed by a rate-1 recursive systematic convolutional code. Simulation results showed that the proposed technique outperforms in terms of bit error rate the codes that exploited one-dimensional (1D) source correlation using the modified BCJR algorithm.

Snasel, V. et al [2] have notified the significance of the setting right parameters for the betterment of turbo codes and the associated applications. They have recommended including meta-heuristic search algorithms to do the task. Further, they have solved the problem using Genetic algorithm (GA) and Differential Evolution (DE), which are renowned optimization algorithms.

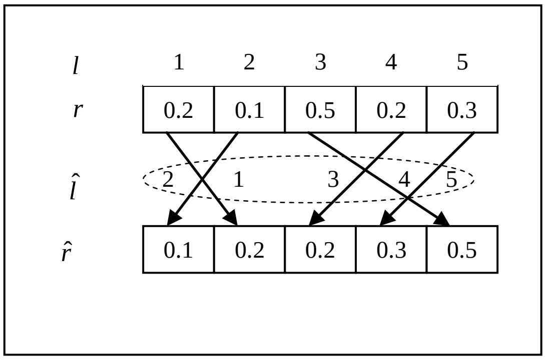


Fig. 5. Encoding Solution Principle

B. Problem Formulation

Most of the methods have concentrated on designing the process that performs the permutation steps. Despite the permutation steps are significant, setting right parameters to the interleavers and the permutation can adequately improve the performance of the turbo encoder. Meta-heuristic search algorithms play key role on it [2]. However, very few works have considered these algorithms for parameter selection. Even the algorithms they have considered are traditional. The traditional algorithms such as GA, DE and Particle Swarm Optimization (PSO) are not robust to handle the current real world problems. Hence, suitable robust algorithm or its variants have to be selected. In addition with it, the developed interleavers are experimentally not demonstrated to work on with real world application such as high dimensional data transmission. While working on with high dimensional data transmission, the traditional way of permutation growth process and the heuristic search process often fails. We consider that introducing a high dimensional heuristic search process can support in designing a high dimensional turbo encoder by robust optimization of interleaver.

3. Proposed Interleaver Design

C. Objective Function

Let us consider an interleaver pattern I_p, which is found to be robust when the $P_r\left(\hat{u}^* = u \mid I_p\right)$ is found to be high. In other words, the symbol error rate $P_b\left(\hat{u}^*, u\right) \mid I_p$ should be minimum. This can be formulated as a minimization function as given in eq. (1), where, I_p^* is the optimized interleaver pattern, $\{I_p\}$ is the set of potential interleaver patterns, $P_b(x, y)$ refers to the error probability between x and y.

$$I_p^* = \underset{\{I_p\}}{\arg\min}\, P_b\left(\hat{u}^*, u\right) \qquad (1)$$

The estimated sequence is obtained from the Log-MAP decoder, which are the maximum a posteriori (MAP) probability algorithm developed in logarithmic domain [21] [22], since the MAP decoders were founds as the optimal turbo decoders [23]. The circuit for turbo decoder is given in fig. 4, where the Jacobian algorithm plays key role by using $\max^*(\;)$, which calculates the logarithm of the sum of exponential terms [24] as detailed in eq. (8), based on the derivations given in eq. (2) – (7). Numerous improvements have been reported in the literature afterwards [25]. However, the MAP based estimation follows the principle of estimating the most probable sequence as given in eq. (9), where $\hat{u}^*$ refers to estimated sequence type and y refers to the received signal.

$$\max^*(x_i : 0 \le i \le n-1) = \ln\left[\sum_{i=0}^{n-1} e^{x_i}\right] \quad (2)$$

$$\max^*(x_1, x_2) = \max(x_1, x_2) + \ln\left(1 + e^{-|x_2 - x_1|}\right) \quad (3)$$

$$\max^*(x_1, x_2) = \max(x_1, x_2) + \ln(|x_2 - x_1|) \quad (4)$$

$$\max_k^*(x) = \max^*(x_{k-1}, x_k) \quad (5)$$

$$\max^*(x_1, x_2, x_3) = \max^*\left(\max^*(x_1, x_2), x_3\right) \quad (6)$$

$$\max_{k=3}^*(x) = \max^*\left(\max_{k=2}^*(x), x_3\right) \quad (7)$$

$$\max_k^*(x) = \max^*\left\lfloor \max_{k-1}^*(x), x_t \right\rfloor \quad (8)$$

$$\hat{u}^* = \underset{\hat{u}}{\arg\max}\, P_r\{\hat{u} \mid y\} \quad (9)$$

D. Hybridization of GA and GSO (Hybrid GSO or HGSO)

1) GSO: It is a kind of optimization algorithm that is population-dependent, wherein the searching activity of the animals is modeled in a mathematical way [26]. Figure 4 portrays the pseudocode of GSO that aids in establishing the interleaver design in the proposed methodology. The three main processes, which constitute the producing operation, are stated below.

- Making a scan at zero degree and conducting random lateral scanning later with the help of three random angles.
- The producer would certainly make a move, only when it discovers a better solution point. Else, it never changes its position.
- When the producer is found to remain at a position for a longer duration without making changes to its position, it moves to zero degree.

Dispersion, the second important processing step, involves the following steps.

- Generation of a random head angle
- Generation of a Gaussian distributed random distance for the purpose of moving towards a new solution point.

Despite the GSO is found to be robust in many applications, it is felt to be incompetent to search in the solution space given in the high dimensional interleaver designing model. Hence, this paper hybridizes GSO with a GA operator for obtaining better performance.

2) GA: During the execution of GA, a population of candidate solutions (also termed as, creatures, individuals, phenotypes) to an optimization problem is found to attain better solutions. Every single candidate solution possesses a group of characteristics, which may be a chromosome or a genotype, and these properties could be subjected to mutation or can undergo a change. The conventional way of representing the solutions is a binary string that is formed of 0s and 1s. Yet, there are also other means of coding [27]. This algorithm works in accordance with the natural theory of evolution [28] for obtaining the near-best solution and it involves three main operators, namely, the selection operator, the crossover operator and the mutation operator. The selection is made in compliance with the concept of the survival of the fittest. On the other hand, the crossover operator makes a recombination of the multiple solutions and the mutation operator allows each and every solution to be kept updated. GA is more thriving because it has wide number of applications till date [29] [30].

3) HGSO: The HGSO of this paper adopts the mutation operator of GA in the GSO so that the exploration problem is well-handled in GSO. The zero degree scanning and lateral scanning in GSO has been adopted from [31]. The outcomes from zero, left and right scanning are given in eq. (10) – (12) , respectively, where, x_p is the parent solution, $\theta_{\max}$ is the maximum pursuit angle, $l_{\max}$ refers to maximum pursuit distance, D_p is the direction of search, r_1 is the normally distributed random integer and r_2 is the uniformly distributed random integer. Under such circumstances, both fine as well as broad searching is enabled based on r_1, though $l_{\max}$, D_p and φ plays role, as per remark 1.

$$x_{zero} = x_p + r_1 l_{\max} D_p \varphi \quad (10)$$

$$x_{left} = x_p + r_1 l_{\max} D_p \left(\varphi - \frac{r_2 \theta_{\max}}{2}\right) \quad (11)$$

$$x_{right} = x_p + r_1 l_{\max} D_p \left(\varphi + \frac{r_2 \theta_{\max}}{2}\right) \quad (12)$$

Remark 1: *When* r_1 *is significantly small, the movement exhibited by scanning is also small. In other words, assume that* $l_{\max}$, D_p and φ *remain constants, whereas* $x_p >> r_1$ (For instance, $x_p > 10$). *Under such circumstances,* $x_{zero} \sim x_p$, i.e., $\left|x_{zero} - x_p\right| \sim 0$.

According to remark 1, the exploration problem of traditional GSO has to be considered while adopting for interleaver design, since the interleaver patterns are essentially distributed. However, the scanning operations are the key strengths of the GSO for quick convergence. Hence, the HGSO modifies the zero scanning procedure using a simple mutation process adopted in GA. The mutation outcome x_{mut}, as given in eq. (13), replaces x_{zero} of GSO and hence the hybrid GSO is derived.

$$x_{mut} = x_{\min} + (x_{\max} - x_{\min})r_1 \qquad (13)$$

Despite numerous mutation techniques have been investigated in the literature [33], this paper adopts the simple mutation to avoid introducing complexity in GSO operation. The remaining procedures of GSO are undisturbed and so the HGSO is aimed for improved performance of GSO and GA.

E. Procedure

1) Encoding: A typical solution usually defines the interleaver pattern, but we adopt a pre-design for interleaver as processing solutions for HGSO. Let us consider an initial solution $x_q : 0 \le q \le N_p - 1$, where N_p refers to the number of possible solutions to be handled by HGSO. The x_q is derived by generating an arbitrary integer set $[r] : r_l \in R; 0 \le l \le L-1$, where L refers to the sum of information and tail bits, followed by sorting it in ascending order and acquiring the index of the sorted order from the actual order, i.e. $x_q = \hat{l}$. The pictorial representation given in fig. 5 shows the process of generating an arbitrary solution having the following properties.

Property 1: *Any element of $[\hat{r}]$ is always is greater than or equal to its preceding element, i.e. $\hat{r}_l \ge \hat{r}_{l-1}$*

Property 2: *The cardinality of $[r]$ and $[\hat{r}]$ is equal to each other, i.e. $|r| = |\hat{r}|$*

Property 3: *l and $\hat{l}$ remain as the indices of the interval $[0, L-1]$, but l can be sequential, while $\hat{l} \in [0, L-1]$*

2) Selection of Producer: Each member x_q is used to solve the objective function given in eq. (1). The member that minimizes the function relatively better than other members is selected as the producer, often represented as x_p.

3) Operators: The other operations are similar to the GSO explained in [26], except the zero scanning process. In HGSO, as stated earlier, the zeros scanning process adopts the mutation operator given in eq. (13).

Remark 2: *The proposed HGSO is considered here as large scale optimization procedure, since it finds the solution of length L (L>10^3)*

TABLE I. SYSTEM AND ALGORITHM CONFIGURATION

Sl. No	Parameter	Configuration
1	Code generator	$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$
2	Code rate	1/2
3	FER_{max}	10
4	N_p	10
5	r1 (HGSO)	0.1
6	E_{max}	100

4. Results and Discussion

The classical turbo encoder – decoder system available in [32] is used as the experimental base. An optimized interleaver is designed using HGSO, GSO and GA and the results are compared against each other along with the performance of random interleaver. The configuration of the system and algorithm are given in Table I. Apart from the parameters, the optimized interleaver is developed with SNR at 0.1 dB, while the objective function is subjected to minimization. The experimentation is carried out on the system with all the aforesaid interleavers for a high dimensional data length of 10^3.

TABLE II. STATISTICAL REPO

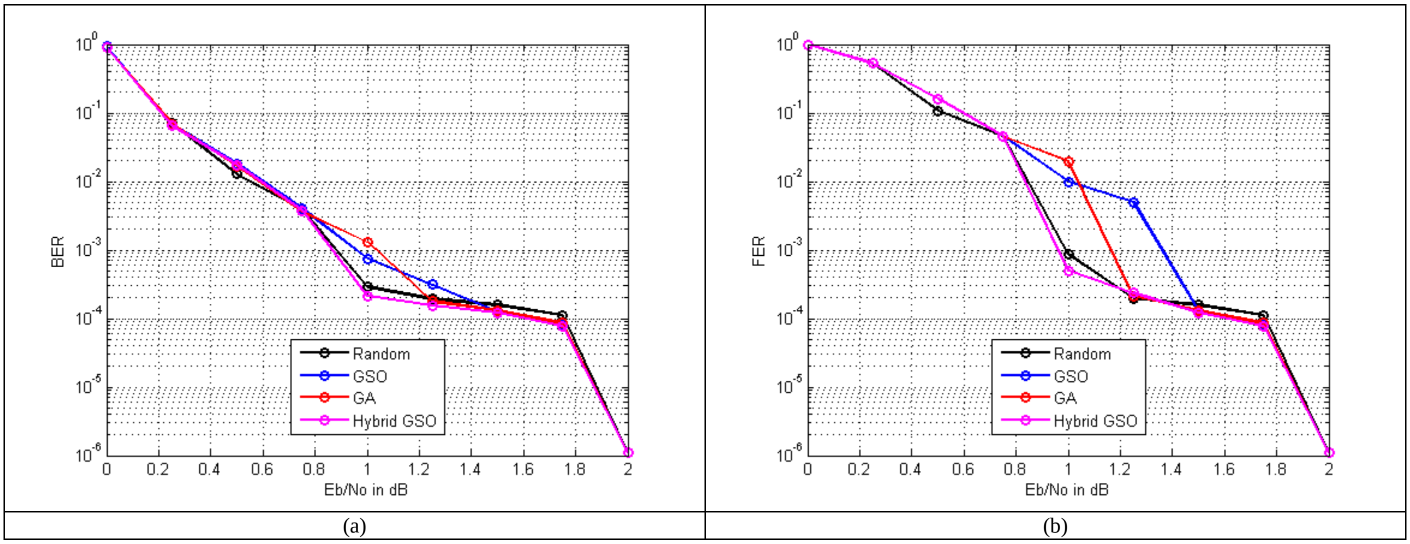
TABLE III. RT ON VARIOUS INTERLEAVER DESIGN METHODS WHEN SUBJECTED TO CASE 2, WHERE FRAME SIZE IS SET AS 2000 BITS

Statistical metric	**Methods**	**E_b/N_o (dB) [Rank]**									**Average Rank**	**Final Rank**
		0.00	*0.25*	*0.50*	*0.75*	*1.00*	*1.25*	*1.50*	*1.75*	*2.00*		
Best BER	Random	9.9x10^{-1} [4]	8.2x10^{-2} [3]	1.1x10^{-3} [1]	3.7x10^{-3} [4]	1.5x10^{-4} [2]	1.8x10^{-4} [3]	0.7x10^{-4} [4]	0.4x10^{-4} [4]	2.4x10^{-5} [3]	3.1	4
	GSO	9.9x10^{-1} [4]	7.9x10^{-2} [2]	2.1x10^{-3} [2]	3.1x10^{-3} [1]	7.1x10^{-4} [3]	2.7x10^{-4} [4]	0.1x10^{-4} [1]	9.6x10^{-5} [1]	2.1x10^{-5} [2]	2.2	2
	GA	9.8x10^{-1} [1]	8.3x10^{-2} [4]	2.3x10^{-3} [4]	3.4x10^{-3} [3]	0.2x10^{-3} [4]	0.9x10^{-4} [2]	0.1x10^{-4} [1]	9.8x10^{-5} [3]	2.7x10^{-5} [4]	2.9	3
	HGSO	9.8x10^{-1} [1]	7.8x10^{-2} [1]	2.1x10^{-3} [2]	3.3x10^{-3} [2]	0.7x10^{-4} [1]	0.2x10^{-4} [1]	0.1x10^{-4} [1]	9.6x10^{-5} [1]	1.8x10^{-5} [1]	1.2	1
Worst BER	Random	9.9x10^{-1} [4]	8.5x10^{-2} [2]	1.7x10^{-3} [1]	5.5x10^{-3} [4]	2.6x10^{-4} [2]	2.4x10^{-4} [3]	0.8x10^{-4} [4]	0.5x10^{-4} [4]	3.8x10^{-5} [4]	3.1	4
	GSO	9.9x10^{-1} [4]	8.6x10^{-2} [3]	2.6x10^{-3} [3]	5.3x10^{-3} [3]	8.1x10^{-4} [3]	3.3x10^{-4} [4]	0.1x10^{-4} [1]	9.7x10^{-5} [2]	2.9x10^{-5} [2]	2.8	3
	GA	9.9x10^{-1} [1]	8.7x10^{-2} [4]	2.5x10^{-3} [2]	4.6x10^{-3} [1]	1.1x10^{-3} [4]	1.3x10^{-4} [2]	0.2x10^{-4} [3]	9.9x10^{-5} [3]	3.6x10^{-5} [3]	2.5	2
	HGSO	9.9x10^{-1} [1]	8.3x10^{-2} [1]	2.6x10^{-3} [3]	5.1x10^{-3} [2]	1.5x10^{-4} [1]	1.2x10^{-4} [1]	0.1x10^{-4} [1]	9.6x10^{-5} [1]	2.8x10^{-5} [1]	1.3	1
Mean BER	Random	9.9x10^{-1} [1]	8.4x10^{-2} [3]	1.3x10^{-3} [1]	4.3x10^{-3} [4]	2.1x10^{-4} [2]	2.2x10^{-4} [3]	0.8x10^{-4} [4]	0.5x10^{-4} [4]	3.5x10^{-5} [4]	2.9	4
	GSO	9.9x10^{-1} [1]	8.2x10^{-2} [2]	2.4x10^{-3} [3]	4.1x10^{-3} [2]	7.6x10^{-4} [3]	3.1x10^{-4} [4]	0.1x10^{-4} [1]	9.7x10^{-5} [2]	2.7x10^{-5} [2]	2.2	2
	GA	9.9x10^{-1} [1]	8.5x10^{-2} [4]	2.3x10^{-3} [2]	3.9x10^{-3} [1]	0.7x10^{-3} [4]	1.1x10^{-4} [2]	0.2x10^{-4} [3]	9.9x10^{-5} [3]	3.4x10^{-5} [3]	2.6	3
	HGSO	9.9x10^{-1} [1]	8.1x10^{-2} [1]	2.4x10^{-3} [3]	4.2x10^{-3} [3]	1.1x10^{-4} [1]	0.7x10^{-4} [1]	0.1x10^{-4} [1]	9.6x10^{-5} [1]	2.6x10^{-5} [1]	1.4	1
Median BER	Random	9.9x10^{-1} [4]	8.3x10^{-2} [3]	1.5x10^{-3} [1]	4.1x10^{-3} [3]	2.2x10^{-4} [2]	2.1x10^{-4} [3]	0.8x10^{-4} [4]	0.4x10^{-4} [4]	3.6x10^{-5} [4]	3.1	4
	GSO	9.9x10^{-1} [4]	8.2x10^{-2} [1]	2.2x10^{-3} [2]	3.9x10^{-3} [1]	7.7x10^{-4} [3]	3.2x10^{-4} [4]	0.1x10^{-4} [1]	9.8x10^{-5} [2]	2.5x10^{-5} [2]	2.2	2
	GA	9.8x10^{-1} [1]	8.4x10^{-2} [4]	2.3x10^{-3} [3]	4.0x10^{-3} [2]	0.8x10^{-3} [4]	1.3x10^{-4} [2]	0.2x10^{-4} [3]	9.9x10^{-5} [3]	3.4x10^{-5} [3]	2.8	3
	HGSO	9.8x10^{-1} [1]	8.2x10^{-2} [1]	2.3x10^{-3} [3]	4.1x10^{-3} [3]	1.2x10^{-4} [1]	0.8x10^{-4} [1]	0.1x10^{-4} [1]	9.6x10^{-5} [1]	2.1x10^{-5} [1]	1.4	1

TABLE IV. STATISTICAL REPORT ON VARIOUS INTERLEAVER DESIGN METHODS WHEN SUBJECTED TO CASE 1, WHERE FRAME SIZE IS SET AS 3000 BITS

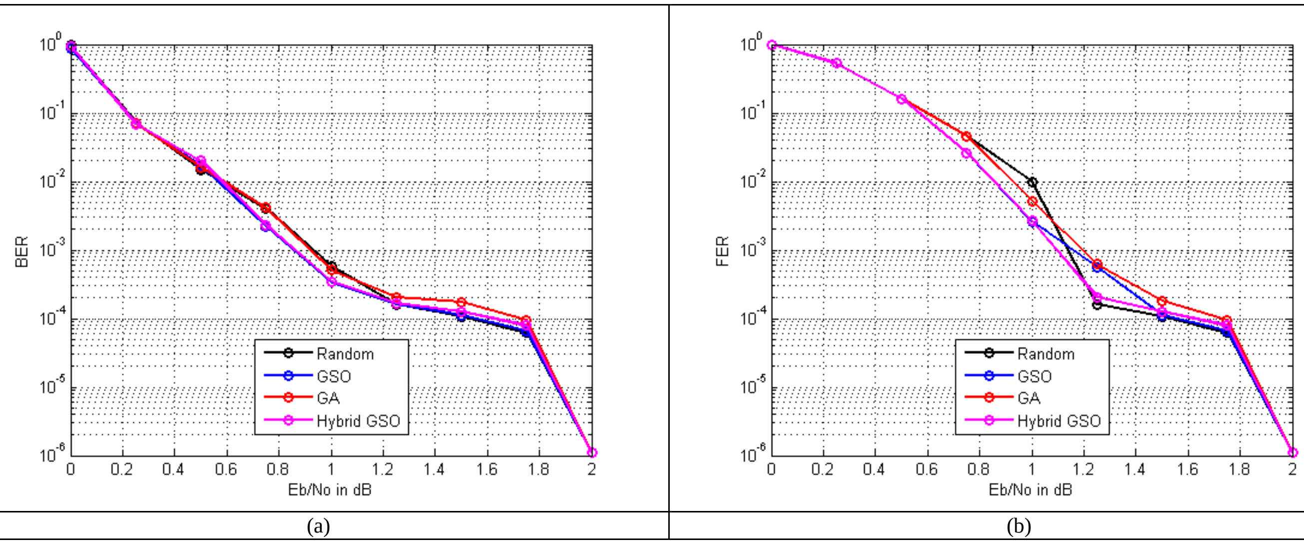
Statistical metric	**Methods**	**E_b/N_o (dB) [Rank]**									**Average Rank**	**Final Rank**
		0.00	*0.25*	*0.50*	*0.75*	*1.00*	*1.25*	*1.50*	*1.75*	*2.00*		
Best BER	Random	9.9x10^{-1} [3]	6.9x10^{-2} [3]	0.6x10^{-2} [1]	3.0x10^{-3} [3]	4.7x10^{-4} [4]	0.5x10^{-4} [1]	0.3x10^{-4} [2]	6.9x10^{-5} [2]	9.9x10^{-6} [1]	2.2	3
	GSO	9.7x10^{-1} [1]	6.9x10^{-2} [3]	0.8x10^{-2} [4]	1.1x10^{-3} [2]	1.6x10^{-4} [2]	0.5x10^{-4} [1]	0.2x10^{-4} [1]	5.9x10^{-5} [1]	9.9x10^{-6} [1]	1.8	2
	GA	9.9x10^{-1} [3]	6.8x10^{-2} [1]	0.7x10^{-2} [2]	3.1x10^{-3} [4]	4.1x10^{-4} [3]	1.0x10^{-4} [4]	0.7x10^{-4} [4]	9.8x10^{-5} [4]	9.9x10^{-6} [1]	2.9	4
	HGSO	9.8x10^{-1} [2]	6.8x10^{-2} [1]	0.7x10^{-2} [2]	1.0x10^{-3} [1]	1.5x10^{-4} [1]	0.5x10^{-4} [1]	0.3x10^{-4} [2]	6.9x10^{-5} [2]	9.9x10^{-6} [1]	1.4	1
Worst BER	Random	9.9x10^{-1} [1]	6.9x10^{-2} [2]	0.9x10^{-2} [1]	3.3x10^{-3} [3]	5.6x10^{-4} [4]	0.8x10^{-4} [2]	0.6x10^{-4} [2]	7.5x10^{-5} [2]	1.9x10^{-5} [2]	2.1	3
	GSO	9.9x10^{-1} [1]	6.9x10^{-2} [2]	1.1x10^{-2} [3]	1.3x10^{-3} [2]	2.5x10^{-4} [2]	0.7x10^{-4} [1]	0.5x10^{-4} [1]	6.4x10^{-5} [1]	2.1x10^{-5} [3]	1.8	1
	GA	9.9x10^{-1} [1]	6.9x10^{-2} [2]	0.9x10^{-2} [1]	3.4x10^{-3} [4]	4.9x10^{-4} [3]	1.1x10^{-4} [4]	1.1x10^{-4} [4]	9.9x10^{-5} [4]	2.1x10^{-5} [3]	2.9	4

	HGSO	9.9x10^{-1} [1]	6.8x10^{-2} [1]	1.3x10^{-2} [4]	1.2x10^{-3} [1]	2.4x10^{-4} [1]	0.8x10^{-4} [2]	0.7x10^{-4} [3]	7.6x10^{-5} [3]	1.1x10^{-5} [1]	1.9	2
Mean BER	Random	9.9x10^{-1} [1]	6.9x10^{-2} [3]	0.7x10^{-2} [1]	3.1x10^{-3} [3]	5.1x10^{-4} [4]	0.7x10^{-4} [3]	0.5x10^{-4} [2]	7.2x10^{-5} [3]	3.9x10^{-5} [4]	2.7	3
	GSO	9.9x10^{-1} [1]	6.9x10^{-2} [3]	0.9x10^{-2} [4]	1.2x10^{-3} [2]	2.1x10^{-4} [1]	0.6x10^{-4} [1]	0.4x10^{-4} [1]	6.2x10^{-5} [1]	3.1x10^{-5} [1]	1.7	2
	GA	9.9x10^{-1} [1]	6.8x10^{-2} [1]	0.8x10^{-2} [2]	3.2x10^{-3} [4]	4.6x10^{-4} [3]	1.1x10^{-4} [4]	0.9x10^{-4} [4]	9.9x10^{-5} [4]	3.8x10^{-5} [3]	2.9	4
	HGSO	9.9x10^{-1} [1]	6.8x10^{-2} [1]	0.8x10^{-2} [2]	1.1x10^{-3} [1]	2.1x10^{-4} [1]	0.6x10^{-4} [1]	0.5x10^{-4} [2]	7.1x10^{-5} [2]	3.1x10^{-5} [1]	1.3	1
Median BER	Random	9.9x10^{-1} [2]	6.9x10^{-2} [3]	0.8x10^{-2} [1]	3.2x10^{-3} [4]	5.2x10^{-4} [4]	0.8x10^{-4} [2]	0.6x10^{-4} [3]	7.3x10^{-5} [3]	3.8x10^{-5} [4]	2.9	4
	GSO	9.8x10^{-1} [1]	6.9x10^{-2} [3]	0.9x10^{-2} [4]	1.1x10^{-3} [1]	2.2x10^{-4} [1]	0.7x10^{-4} [1]	0.5x10^{-4} [1]	6.2x10^{-5} [1]	3.2x10^{-5} [1]	1.6	2
	GA	9.9x10^{-1} [2]	6.8x10^{-2} [1]	0.8x10^{-2} [1]	3.1x10^{-3} [3]	4.5x10^{-4} [3]	1.0x10^{-4} [4]	0.9x10^{-4} [4]	9.9x10^{-5} [4]	3.7x10^{-5} [3]	2.8	3
	HGSO	9.9x10^{-1} [2]	6.8x10^{-2} [1]	0.8x10^{-2} [1]	1.2x10^{-3} [2]	2.2x10^{-4} [1]	0.8x10^{-4} [2]	0.5x10^{-4} [1]	7.2x10^{-5} [2]	3.2x10^{-5} [1]	1.4	1



(a) (b)

Fig. 6. (a) BER and (b) FER plots for case 2: Frame size = 2000 bits



(a) (b)

Fig. 7. (a) BER and (b) FER plots for case 3: Frame size = 3000 bits

Table II states that there is no algorithm to consistently gain the same rank throughout all the SNR variations. However to conclude the performance, the average of all the ranks throughout the SNR variation is determined and a final rank is allotted based on the average rank. HGSO secures first rank in all the statistical metrics, whereas GSO gains second rank except the median BER performance. GA dominates GSO in the median performance and secures third rank for the other metrics. The random interleaver has been identified as the poor performer among all the algorithms.

As per Table III, HGSO based interleaver and random interleaver grabs the first and last position, respectively in all the scenarios. In the worst case scenario, GSO - based interleaver fails to secure second position, though it remains second in the other scenario. Table IV reveals different results from Table II and III. Under best case performance, GA - based interleaver could hold the last position, while the random interleaver gets third position. Similarly, the worst case scenari reveals that the GSO - based interleaver has outperformed all the other inteleavers, while the HGSO - based interleaver gets second position. This interprets that HGSO has a minor uncertainty, yet it can be neglected as it has produced convincable results in all the other circumstances. Moreover, HGSO based interleaver has gained first position at 11 out of 12 instances (= four statistical metrics × three experimental cases). Hence, it can be claimed that the HGSO based interleaver is suitable for high dimensional data transmission.

5. Conclusion

This paper presented a high dimensional interleaver design procedure using hybrid GSO, a Meta – heuristic search algorithm. The algorithm has been developed by hybridizing renowned GSO and GA in high dimensional space and so the turbo encoder exhibits high dimensional data transmission. Reliable test platform has been selected for experimentation in which the performance of HGSO has been compared against GSO, GA and random interleaver design. The BER and FER convergence have demonstrated the acute minimization of error probabilities by HGSO than the other methods. The statistical report has also revealed that the reliability of accomplishing the BER and FER by HGSO is higher than other methods.

References

[1][1] Laddomada, M.; Scanavino, B., “A cost-function based technique for design of good prunable interleavers for turbo codes”, IEEE Transactions on Wireless Communications, Vol. 5, No. 8, pp. 1953 - 1958, 2006

[2] Snasel, V.; Platos, J.; Kromer, P.; Ouddane, N.; Abraham, A., “Interleaver optimization by population based metaheuristics”, 2010 Second World Congress on Nature and Biologically Inspired Computing (NaBIC), pp. 672 - 677, 2010

[3]He, S.; Wu, Q.H. ; Saunders, J.R., “Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior”, IEEE Transactions on Evolutionary Computation, Vol: 13, No: 5, Page(s): 973 – 990, 2009

[4]Ajith Abraham, Pavel Krömer, Václav Snášel, Nabil Ouddane, "Evolving Turbo Code Interleavers by Genetic

Algorithms ", in proceedings of International Conference on Complex, Intelligent and Software Intensive Systems, 2008.

[5]K.Prabhakar, Ch.Ravi Kumar, K.Padmaraju, "Interleaver optimization by particle swarm Algorithm ", International Journal for Technological Research in Engineering, Vol. 2, no.1, September-2014.

[6]Hokfelt J, Edfors O, Maseng T. Interleaver design for turbo codes based on the performance of iterative decoding. In: Proc IEEE ICC 1993. p. 93–7.

[7]Divsalar D, Pollara F. Turbo codes for deep-space communications. Jet propulsion lab, TDA progress report feb. 1995. p. 42–120.

[8]Perez C, Seghers J, Costello D. A distance spectrum interpretation of turbo codes. IEEE Trans Inf Theory 1996;42(2):1698–703.

[9]Dolinar S, Divsallar D. Weight distribution of turbo codes using random and nonrandom permutations. Jet propulsion lab, TDA progress report aug. 1995. p. 42–122.

[10] Sadjadpour HR, Sloane NJA, Salehi M, Nebe G. Interleaver design for turbo codes. IEEE J Sel Areas Commun 2001;19(5):831–7.

[11] Crozier N. New high-spread high-distance interleavers for turbo codes. In: Proc 20th biennial symp commum 2000. p. 3–7.

[12] Simona V. Halunga, Nicolae Vizireanu, Octavian Fratu, "Imperfect cross-correlation and amplitude balance effects on conventional multiuser decoder with turbo encoding", Digital Signal Processing, vol. 20, pp. 191–200, 2010.

[13] Muhammad Arif, Noor Muhammad Sheikh, Asrar U.H. Sheikh, "Design of two step deterministic interleaver for turbo codes", Computers and Electrical Engineering, vol. 34, pp. 368–377, 2008.

[14] Lucian Trifina & Daniela Tarniceriu, "Improved method for searching interleavers from a certain set using Garello's method with applications for the LTE standard", annals of telecommunications - annales des télécommunications, Vol. 69, no. 5-6, pp 251-272, 2014.

[15] Horia Balta, Catherine Douillard and Radu Lucaciu, "Multi-non-binary turbo codes", EURASIP Journal on Wireless Communications and Networking, 2013.

[16] Mohd Azri Mohd Izhar, Norsheila Fisal, Xiaobo Zhou, Khoirul Anwar and Tad Matsumoto, "Exploitation of 2D binary source correlation using turbo block codes with fine-tuning", EURASIP Journal on Wireless Communications and Networking, 2013.

[17] C. Berrou and A. Glavieux, "Near optimum error correcting coding and decoding: Turbo-codes," IEEE Trans. Commun., vol. 44, no. 10, pp. 1261-1271, 1996.

[18] C. Berrou, R. Pyndiah, P. Adde, C. Pouillard, and R.L. Bidan, "An overview of Turbo codes and their applications," in The European Conference on Wireless Technology, Paris, Oct. 2005, pp. 1-9.

[19] F. Peper, "Nanocomputers," in Computational Complexity, New York: Springer, 2012, pp. 1998-2028.

[20] Design and Simulation of Turbo Encoder in Quantum-Dot Cellular Automata Zhang, M.; Cai, L.; Yang, X.; Cui, H.; Feng, C. Nanotechnology, IEEE Transactions on Year: 2015, Volume: 14, Issue: 5 Pages: 820 - 828, DOI: 10.1109/TNANO.2015.2449663

[21] 4. P Robertson, E Villebrun, P Hoeher, A comparison of optimal and sub-optimal MAP decoding algorithms operating in the log domain, in Proc. IEEE International Conference on Communications (ICC) (Seattle, USA, 1995), pp. 1009–1013

[22] 5. P Robertson, P Hoeher, E Villebrun, Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding. ETT 8(2), 119–125 (1997)

[23] 3. LR Bahl, J Cocke, F Jelinek, J Raviv, Optimal decoding of linear codes for minimizing symbol error rate. IEEE Trans. Inform. Theory IT-20(2), 284–287 (1974)

[24] AJ Viterbi, An intuitive justification and a simplified implementation of the MAP decoder for convolutional codes. IEEE J. Sel. Areas Commun. 16(2), 260–264 (1998)

[25] Michal Sybis and Piotr Tyczka, "Reduced complexity Log-MAP algorithm with Jensen inequality based non-recursive max$*$ operator for turbo TCM decoding", EURASIP Journal on Wireless Communications and Networking 2013, 2013:238

[26] He, S.; Wu, Q.H. ; Saunders, J.R., "Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior", IEEE Transactions on Evolutionary Computation, Vol: 13, No: 5, Page(s): 973 – 990, 2009

[27] Goldberg, David (1989). Genetic Algorithms in Search, Optimization and Machine Learning. Reading, MA: Addison-Wesley Professional. ISBN 978-0201157673.

[28] Holland, John (1992). Adaptation in Natural and Artificial Systems. Cambridge, MA: MIT Press. ISBN 978-0262581110.

[29] Aloysius George, B. R. Rajakumar, D. Binu,(2012) "Genetic algorithm based airlines booking terminal open/close decision system". ICACCI '12 Proceedings of the International Conference on Advances in Computing, Communications and Informatics Pages 174-179

[30] Zhang, J., Lo, W.L., and Chung, H., "Pseudocoevolutionary Genetic Algorithms for Power Electronic Circuits Optimization", IEEE Trans Systems, Man, and Cybernetics, Part C., Vol.36, No.4, July 2006, pp. 590–598.

[31] W. J. O'Brien, B. I. Evans, and G. L. Howick, "A new view of the predation cycle of a planktivorous fish, white crappie (pomoxis annularis)," Canadian J. Fisheries Aquatic Sci., vol. 43, pp. 1894–1899, 1986

[32] Won Y. Yang, Yong S. Cho, Won G. Jeon, Jeong W. Lee, Jong H. Paik, Jae K. Kim, Mi-Hyun Lee, Kyu I. Lee, Kyung W. Park, Kyung S. Woo, MATLAB®/Simulink® for Digital Communication, A-Jin Publishing Co., Korea, 2009

B.R. Rajakumar, A George, "A new adaptive mutation technique for genetic algorithm", 2012 IEEE Computational Intelligence & Computing Research (ICCIC), 2012

1st author: Ms. Rutuja Deshmukh
Research Scholar, Electronics Department, Priyadarshini College Of Engineering, Hingna Nagpur-440019

I have completed BE in Electronics and Telecommunucation Engineering from Raisoni College of Engineering Nagpur. I have done M.Tech in VLSI from Priyadarshini College of Engineering Nagpur. Previously I have published 2 papers in International Journals.Presently I am working as an Assistant Professor at D.Y. Patil College of Engineering Pune. I have worked as an organizer of National Conference held on 9th-10th Jan, 2015. I have also worked as an organizer of Faculty development Program held at DYPCOE, Pune. I have an experience of 4.5 years in teaching field.

2nd Author **Dr Ashish R. Panat,BE, ME, PhD(Electronic Engineering) 29 Years of Experience in Academics 3 Patents filed, Over 40 Pubications. Research Guide** Priyadarshini College of Engineering Nagpur