Computers in Biology and Medicine 165 (2023) 107404

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

BO-DNA: Biologically optimized encoding model for a highly-reliable DNA data storage

Abdur Rasool [a,b], Jingwei Hong [a,c], Qingshan Jiang [a,*], Hui Chen [d], Qiang Qu [a,**]

[a] Shenzhen Key Laboratory for High Performance Data Mining, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, 518055, China
[b] Shenzhen College of Advanced Technology, University of Chinese Academy of Sciences, Beijing, 100049, China
[c] College of Mathematics and Information Science, Hebei University, Baoding, 071002, China
[d] Shenzhen Polytechnic University, Shenzhen, 518055, Guangdong, China

ARTICLE INFO

Keywords:
Bio-constrained codes
Optimized encoding
DNA data storage
Biocomputing
Reliable storage

ABSTRACT

DNA data storage is a promising technology that utilizes computer simulation, and synthetic biology, offering high-density and reliable digital information storage. It is challenging to store massive data in a small amount of DNA without losing the original data since nonspecific hybridization errors occur frequently and severely affect the reliability of stored data. This study proposes a novel biologically optimized encoding model for DNA data storage (BO-DNA) to overcome the reliability problem. BO-DNA model is developed by a new rule-based mapping method to avoid data drop during the transcoding of binary data to premier nucleotides. A customized optimization algorithm based on a tent chaotic map is applied to maximize the lower bounds that help to minimize the nonspecific hybridization errors. The robustness of BO-DNA is computed by four bio-constraints to confirm the reliability of newly generated DNA sequences. Experimentally, different medical images are encoded and decoded successfully with 12%–59% improved lower bounds and optimally constrained-based DNA sequences reported with 1.77bit/nt average density. BO-DNA's results demonstrate substantial advantages in constructing reliable DNA data storage.

1. Introduction

DNA data storage is the latest technology with the potential to store large amounts of data in a small, stable, and durable format, offering benefits such as high data density and long-term stability. Reliability and stability are crucial to ensure the stored data remains intact [1–3]. In this technology, computer simulations are executed to convert digital files into binary data (0, 1) which are encoded into the DNA sequences database of A, C, G, and T nucleotides. This database is used to synthesize DNA in a wet lab, and data is stored in DNA. The synthesized DNA is sequenced to retrieve the original binary data by decoding simulations [4–7]. This process is illustrated in Fig. 1.

Computer simulations play a pivotal role in facilitating the design of encoding models for converting binary data into oligonucleotides, thereby enabling the synthesis of DNA, which constitutes a fundamental aspect of synthetic and molecular biology. During synthesizing and sequencing DNA, minimizing the amount of DNA required to store a given amount of data is essential. It requires that the data be compressed and encoded in an optimized way for the specific limitations of DNA storage. For instance, DNA sequences are limited in length, meaning large files must be stored in smaller fragments. For which an optimized encoding scheme can generate to overcome this limitation [8]. Additionally, errors occur during the process of synthesizing and sequencing DNA, e.g., nonspecific hybridization errors refer to the binding of a synthetic oligonucleotide to a DNA molecule that is not the intended target, resulting in the incorrect readout of the stored information [9]. It can occur due to similar or complementary sequences in the stored DNA or variations in the DNA codewords. These errors can crucially affect the reliability of the stored data. To minimize these errors, it is crucial to construct the DNA codes carefully, for which.

primary biological coding constraints (GC content, Homopolymer) are implemented with an encoding mechanism [10]. These mechanisms can reduce the number of errors by using error correction codes or computational methods and ensuring that the data is robust against the types of errors that are likely to occur in DNA storage. One of the computational methods is heuristic computation, which comprises a set

* Corresponding author.
** Corresponding author.
E-mail addresses: rasool@siat.ac.cn (A. Rasool), qs.jiang@siat.ac.cn (Q. Jiang), qiang@siat.ac.cn (Q. Qu).

https://doi.org/10.1016/j.compbiomed.2023.107404
Received 21 March 2023; Received in revised form 13 August 2023; Accepted 26 August 2023
Available online 28 August 2023

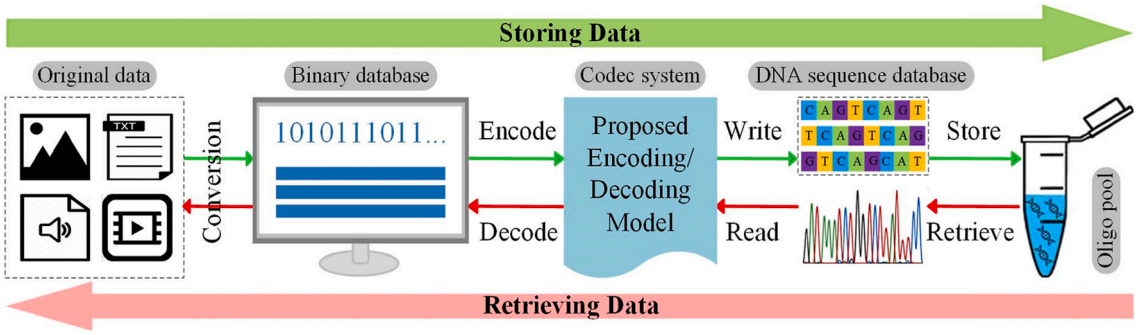

Fig. 1. An overview of an entire DNA data storage system with its major stages.

of biologically-inspired optimization algorithms [9,11,12]. In synthetic biology, optimization algorithms can design DNA sequences for various purposes, i.e., lower-bounds improvements in DNA code generation. These algorithms generate a population of candidate DNA sequences, which are then subjected to various evaluation criteria. The best-performing sequences are then selected, recombined, and mutated to form a new generation of candidates. This process is repeated until an optimal DNA sequence meets the desired criteria [13].

Recently, various literature [9,12,14] has been reported utilizing optimization algorithms for optimal DNA sequence construction. For instance, a damping multi-verse optimizer [9] was reported to design a set of DNA coding as non-payload and obtained a larger set of coding. The authors [14] developed an optimization-based algorithm to reduce the nonspecific hybridization errors for constructing lower bounds of DNA codes. In our recent prior research [15], a heuristic-based moth--flame optimization was applied for the DNA code construction, enhancing the lower bounds. These bounds can be used to determine the minimum number of nucleotides required to store a given amount of data with a given error correction capability. These studies demonstrated the algorithm's efficiency in practical optimization, and lower bounds results were equal to or surpassed the prior works of DNA codes. However, the diversity of the sample populations was still lacking. As a result, there was a large difference in sequence length, and the presence of similar or complementary sequences was also an obstacle to synthesizing reliable DNA. Meanwhile, there was a crucial gap; the lack of DNA codes implementation in a real scenario for encoding/decoding the digital data with an optimization-based encoding method.

The reason for designing the proposed biologically optimized encoding method is to implement the heuristic algorithm in a real case of encoding/decoding of information. As we know DNA synthesis process relies on the DNA codes or sequences, which often have incorrect base pairing due to minimum lower bounds and unconstrained sequences [16]. For reliable DNA data storage, DNA codes must have maximum lower bounds, which can help to minimize nonspecific hybridization errors [2]. For this, we continue optimizing coding over DNA sequences by computer simulations. The proposed optimized encodings played a crucial role in constructing reliable DNA codes by reporting the efficiency, accuracy, and stability of the mapping scheme to translate digital data into DNA sequences. It can help to make DNA data storage a more practical and reliable solution for the long-term preservation of digital information.

In this paper, a novel rule-based mapping method is designed to implement the biologically optimized encoding for converting the binary data to DNA nucleotides (BO-DNA). The previous moth-flame optimization algorithm [17] is customized (CMF) to initialize the sample population by chaotic mapping to maximize lower bounds and satisfy the bio-constraints. The experiments used medical image data with different extensions to achieve optimized encoding efficiency and compare with other files and benchmark studies. To the best of our knowledge, this is the first work to open the door for the practical implementation of optimization in DNA data storage technology. The data and codes for this work are available at https://github.com/abdul-rasool/BO-DNA.

The following are the main contributions.

1. A novel rule-based mapping method is introduced by different base types to avoid data drop during the transcoding of binary data to premier nucleotides for reliable DNA data storage.
2. A tent chaotic map initializes the diverse sample population of premier DNA sequences, and a customized optimization algorithm is constructed with maximum iteration to balance the exploitation and exploration abilities of the proposed encoding. Further, new lower bounds are derived on the optimized DNA codes to control nonspecific hybridization errors.
3. Additionally, a computational bio-constrained threshold is reported to ensure the DNA sequence's reliability by meeting the four coding constraints. Experimentally different medical images are encoded and decoded successfully with improved density, demonstrating the importance of BO-DNA for reliable DNA data storage.

The structure of the remaining paper is as follows: Section 2 elaborates on the literature work on existing encoding methods and optimization role in DNA storage. Section 3 introduces the proposed biological optimized encoding, Section 4 delivers experiments and results evaluations, Section 5 provides the discussion and limitations, and Section 6 concludes this study.

2. Related work

2.1. Encodings of DNA data storage

The first concept of DNA data storage was proposed by Davis [18] in 1996. DNA storage technology is progressing slowly because of the limitations of synthesis and sequencing technologies. Since this century has powerful new tools for synthesizing and sequencing DNA, the topic of DNA storage has been widely studied. Numerous researchers have investigated the possibility of storing digital data in DNA with different encoding methods and models. In 2012, Church [4] created arbitrary encoding to store 0.65 MB files in DNA. To ensure the accuracy and integrity of the stored data, Yazdi [19] proposed an effective storage architecture in 2015 that appended specific unique address bits of 20 bps length to the ends of a 1000 bps data block to store the Wikipedia of six universities. An effective and reliable forward error correction scheme was created in 2016 by Blawat [20] that can handle mistakes in DNA synthesis, sequencing, etc. These codec approaches shows that DNA can function as a durable storage in the future.

In 2017, the Fountain-based coding method was used by Erlich [5] to create concise and convincing DNA encoding schemes. Their encoding produces different numbers of DNA bases to receive maximum tunable redundancy without obscuring the protocol structure. Encoding eliminates incorrect oligonucleotides, conserving high-quality sequencing fragments for highly reliable encoding and decoding. In 2018, Organick

[7] proposed an encoding method that decreased the likelihood of sequence errors in the DNA during synthesis and sequencing and achieved a high storage density. The author [21] designed HEDGES (hash encoded, decoded by greedy exhaustive search) coding for error correction by satisfying the biological coding constraints (GC content and no-runlength). To cope with the sequence length with constraints, Schwarz [22] proposed near-optimal rateless erasure codes (NORECs) based on Huffman Encoding [23] for high-capacity DNA storage. In 2022, Ping [24] proposed the yin-yang codec, a powerful transcoding algorithm that uses two rules to create DNA sequences highly compatible with synthesis and sequencing technologies. This year (2023), a study presented a concatenated coding scheme to generate variable-sized encoded sequences with user-defined coding constraints (GC content and Homopolymer) to avoid undesirable motifs [10].

There is no single coding method that is universally considered to be the best for reliable DNA data storage. Different encoding methods have their advantages and limitations, and the choice of coding method for DNA data storage depends on the precise requirements of the application and the trade-offs between storage efficiency and implementation complexity. For example, fixed-length encoding [25] divides the binary data into fixed-length blocks. It encodes each block into DNA nucleotides using a fixed mapping scheme, e.g., low-density parity-check coding [26]. In contrast, the arithmetic encoding model [10] is similar to Huffman encoding [23], but it offers efficient compression of data, e.g., adaptive encoding [27]. Although DNA as a storage medium has a high theoretical storage density, it is challenging to reach the theoretical limit due to sequencing and synthesis errors in DNA storage channels and biochemical constraints on sequences. As a result, much work has gone into optimizing coding techniques that can reduce errors and information loss for reliable DNA data storage.

2.2. Optimization role in DNA storage

Since the studies mentioned earlier had a significant role in the development of DNA data storage, heuristic-based optimization algorithms were also able to solve various problems in the simplest possible ways. In 2011, the DNA sequence was compressed by a novel adaptive particle swarm memetic algorithm. It optimized the DNA code repeat in a DNA sequence [13]. Later, the advancement of optimization algorithms was reported in various studies [11,28,29]. In 2020, HEDGES encoding used the optimization-based greedy exhaustive search method to balance the GC content ratio in DNA sequences [21]. The author [14] utilized a Brownian-based multi-verse optimizer to reduce the error rate by minimum free energy constraints by constructing DNA coding sets. In 2022, Cao [27] introduced an adaptive encoding mechanism by considering various coding regions based on a different optimization algorithm, i.e., DMVO [9], and various DNA coding constraints, i.e., GC content, Hamming distance, no-runlength. This encoding was based on the Fountain code method and used Gibson assembly address bits to store the data. These optimization studies were synergized by mutation strategies for effective convergence [15]. However, their initial population convergence was adversity-affected due to these strategies. In contrast, the variation in the initial population significantly impacts convergence effectiveness and the quality of the optimal global solution. Meanwhile, these studies were only creating the DNA codes for data storage, but there is still a crucial gap in the practical implementation of these created DNA codes with novel encoding methods satisfying the biological coding constraints.

3. Proposed optimized encoding: BO-DNA

This paper proposed the mapping rules, which are optimized with a heuristic algorithm for constructing high-reliable and stable DNA sequences. Hereafter, the proposed mechanism is BO-DNA (Biologically Optimized encoding model for DNA). The importance of biologically optimized encoding is given in Section 1. This encoding is based on our proposed mapping rules, chaotic-based customized moth-flame optimizer (CMF), and biological combinatory constraints. In this regard, BO-DNA's significant contribution can be categorized into three following stages.

1. *DNA mapping rules:* Conversion of digital data into binary bits and mapping these bits into premier DNA nucleotides by our proposed mapping rules,
2. *Optimization of encoding:* Utilization of tent chaotic with a customized optimizer CMF for the improvement of lower bounds of DNA sequences with maximum iteration,
3. *Bio-coding constraints:* Calculation of biological combinatorial constraints in order to generate optimal and reliable DNA sequences.

The comprehensive details of these substantial steps are presented in the following subsections. BO-DNA's process diagram illustrates the schematic connection between these steps (Fig. 2). The digital data is converted into a binary format using the Python function. The binary data is divided into N groups based on their size, and then each group is divided into n segments based on the group size. Each data group can therefore be viewed independently, making it easy to read by the proposed mapping rules.

3.1. Proposed mapping rules

The inspiration of our mapping rules is based on Fountain encoding [5], that is, {00, 01, 10, 11} $\leftrightarrow$ $\{A, T, C, G\}$ scheme. In the Fountain method, this mapping is applied after the XORed function, and basic coding constraints (GC content and Homopolymers) are used to judge sequence satisfaction criteria. The sequences that do not satisfy these constraints are directly discarded, which loses the portion of binary segments and causes the original information reliability. Therefore, we have designed an optimized encoding scheme to avoid data drops after the basic mapping.

The binary segments are read through the encoder based on proposed mapping rules distributed in Tables 1 and 2. According to the number of base repeats in the DNA sequence, tandem repeats are divided into single and two base types. A step-by-step functionality of each Table and the base type is elaborated as follows.

1. The single-base repeat sequences are unstable in organisms and often mutate in replication and transcription. DNA fragments with single base repeats are prone to base mutation and deletion. In DNA data storage, copying and amplifying DNA sequences is necessary. In order to realize the reliable storage of DNA sequences and avoid initial codons, this paper constructs a single base repeat sequence base library $S_1 \in \{AAA, TTT, CCC, GGG\}$ and $S_2 \in \{ATG, TAA, TGA, TAG, TGG\}$ to avoid the above base sequences when performing encoding. This mapping has three stages; first 4 bits, intermediate bits, and last bit or 2 bits.
2. After converting the first 4 binary bits to DNA bases by utilizing the Fountain encoding scheme, it is necessary to avoid the base sequence binding with the single base repeat in the base library of DNA. When the current base pair is converted typically, it is essential to construct a two-bit base set, $B = \{AT, TA, TG, TT, AA, GG, CC\}$, given in Table 1. The encoder reads the next binary bit from the intermediate bits, corresponding to the prior base pair, and maps it with a single base. It continually scans whether the base formed by each generated base and the previous base is not repeated. The base conversion of the DNA sequence shall be carried out accordingly.
3. Meanwhile, if the next 1 binary bit, with a prior base pair, does not exist in set B, the encoder considers the 2 consecutive binary bits and maps accordingly. However, if the previous base pair with the next 2 bits do not exist in set B, but the corresponding bits are in X; then map with a single base. In order to generate reliable DNA codes by

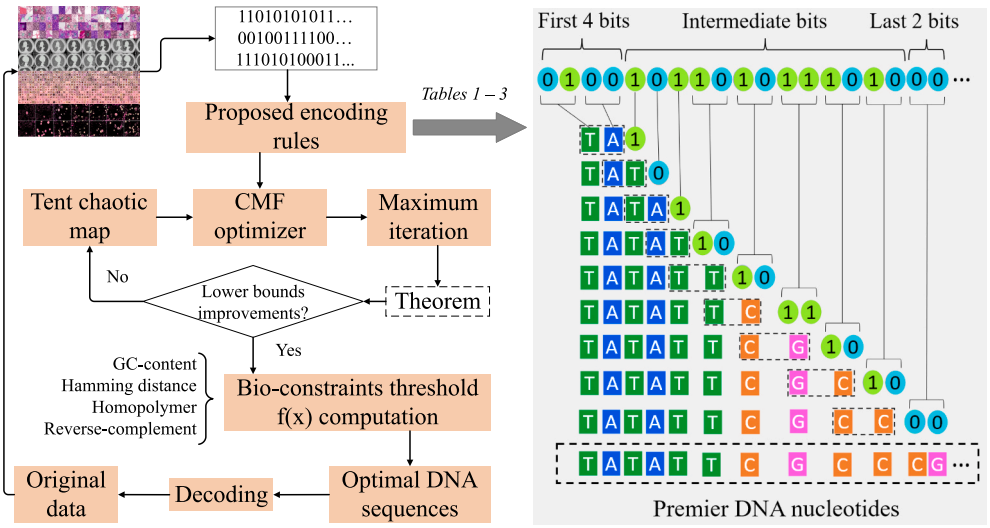

Fig. 2. The schematic diagram proposed biological optimized encoding model, BO-DNA, encodes the digital data into optimal DNA sequences for data storage.

Table 1
Mapping of intermediate bits to nucleotides.

	AT	TA	TG	TT	AA	GG	CC	X
A	0	–	–	0	–	0	0	00
T	10	1	1	–	10	10	10	01
C	11	0	0	10	0	11	–	10
G	–	–	–	11	11	–	11	11

Table 2
Mapping of the last one bit or 2 bits to a pair of nucleotides.

0	1	00	01	10	11
AC	TC	CG	GA	GT	GC

avoiding two base tandem repeats, some binding sites are left empty with the "-" mark in Table 1.

4. The encoder is trained to read the last binary bit or two bits shall be mapped with 2 bases by Table 2.

These rules aid in transferring DNA sequences into cells to realize natural biological storage. In this way, it is necessary to put forward more restrictive requirements for the generated DNA sequences: the most critical point is to minimize the interference of these exogenous DNA sequences on the normal life activities of cells. Suppose the possibility of these randomly arranged exogenous DNA being translated into protein in the cell can be minimized. In that case, the interference of this exogenous DNA to the cell life activities would also be restrained to a certain extent. We know that the translation of proteins in cells depends on the triplet codon sequence of genes and messenger RNA. The initial codon is the first triplet codon on messenger RNA that is translated by the ribosome. Therefore, we construct a set of initial codons $S_1 \in$ {ATG, TAA, TGA, TAG, TGG}, hoping that the transformed DNA sequence can avoid the occurrence of initial codons.

A random binary sequence is mapped into premier DNA codes according to rule-based mapping in Fig. 2 to demonstrate the mapping process as an example. Similarly, random binary segments 01.11.0.00.00.11 and 10.11.10.10.0.0 1.10.0.0.11.01.10.10 are encoded into DNA sequence fragment T.G.C.A.A.GC and CG.C.C.A.T.T.A.C.G.T.C.GT, respectively, and decoded accurately. Dots are added among corresponding bits and bases for more clarity and easy understating of mapping using proposed rules. The construction of premier DNA sequences through the proposed DNA mapping rules is structured in Algorithm 1.

Algorithm 1. DNA storage code set construction algorithm satisfying constraints.

Algorithm 1. DNA storage code set construction algorithm satisfying constraints.

Input: Mapping of first four bits b_4, intermediate bits Int_b and last bit l_b with respect to (w.r.t) Tables 1 or 2 (T1, T2), exclude set S and prior 2 bases $2B$.

1: mapping initialization;
2: convert first 4 bits b_4 as per Fountain
3: $Int_b = 5$ // Initial value of Int_b
4: **while** ($b_4 \leqslant Int_b < l_b$) // mapping Int_b
5: **if** (2B and Int_b in T1) **then**
6: map Int_b w.r.t prior 2 bases from T1
7: **else if** (2B and Int_b are not in T1) **then**
8: read two consecutives Int_b
9: map two Int_b w.r.t prior 2 bases from T1 //Map two Int_b with respect to prior 2 bases
10: **else if** (2B $\notin$ T1, but $Int_b \in$ S) **then**
11: map two bits w.r.t 2B and X from T1
12: **end if**
13: **end while**
14: **if** ($Int_b < l_b$) **then**
15: **for** ($i = Int_b$; $i < l_b$; $i++$) **do** // Starting and ending values for the nested loops
16: encode the last bit or two bits as per T2
17: **end for**
18: **end if**

Output: DNA sequences.

In order to determine the position of each DNA sequence fragment and meet the requirements of random reading, two reserved marker bases (RB) are used to represent the information of sequence number in the DNA sequence segment, i.e., ATTATT represents the fragment

Table 3
DNA sequence fragment number and corresponding reserved base.

F#	0	1	2	3
RB	ATTATT	ATTAAG	ATTCCG	AAGATT
F#	4	5	6	7
RB	AAGAAG	AAGCCG	CCGATT	CCGAAG

number (F#) "0″, ATTAAG indicates the F# "1″, etc. The corresponding relationship between the reserved base (RB) and the F# is given in Table 3. After assigning the index to each DNA sequence related to each data, 8 DNA fragments are taken into account from each data. It is convenient to know whether it is necessary to re-read DNA sequence segments that are not fully replicated or inaccurate in DNA sequence sequencing data. It helps to achieve an orderly reading of DNA sequence segments stored in DNA data. When DNA data storage is carried out to restore DNA sequence files to the computer, each stored data block can be directly and quickly located by the serial number represented by the reserved base information.

The proposed mapping rules are defined to construct the premier DNA sequences, which enhance the density and partially control the coding constraints at this stage. However, the restricted bio-constraint satisfaction is based on the optimized DNA sequence generated by the customized optimization algorithm (CMF).

3.2. Optimization of encoding

The premier DNA sequence is processed to be optimized through the moth-flame optimization (MFO) algorithm due to its diverse application in compression mechanisms [29]. Despite receiving a high convergence rate, MFO is still unable to locate global optima that might improve the algorithm's convergence rate. Therefore, the customized algorithm is created by incorporating Chaos [28] into the MFO algorithm to lessen this effect and increase its efficiency, thereby termed the CMF algorithm. When an algorithm exhibits chaotic behavior, it can correlate with a specific parameter using a function, representing the quality of a complex system with unpredictable responses. Compared to stochastic searches, which primarily rely on probabilities, chaos may conduct overall searches at higher speeds due to its ergodicity and non-repetition features. The conventional initialization approach in MFO may not ensure randomness and diversity in the initial population. However, tent chaotic mapping as an initialization method offers both order and ergodicity. By employing tent chaotic mapping for population initialization, this study effectively preserves the variety of the population. The mathematical model is presented in Eq. (1) [30]:

$$x_{i+1,j} = \begin{cases} 2x_i, 0 \le x_i \le 0.5 \\ 2(1-x_i), 0.5 < x_i \le 1 \end{cases} \tag{1}$$

where x is a random number ranging from 0 to 1 with j constant, as the tent chaotic sequence contains a number of modest periodic points x_i as well as unstable periodic points x_{i+1}. A random variable called *rand* $(0,1)/N_T$ is added to the tent chaotic map in order to avoid these locations and keep the chaotic sequence's features. The revised tent chaotic map is represented by the following equation:

$$x_{i+1,j} = \begin{cases} 2x_i + \dfrac{rand(0,1)}{N_T}, 0 \le x_i \le 0.5 \\ 2 - x_i + \dfrac{rand(0,1)}{N_T}, 0.5 < x_i \le 1 \end{cases} \tag{2}$$

where N_T denotes the number of participants (population) in the chaotic sequence of tents.

The revised tent chaotic map (Eq. (2)) is utilized for initializing the i th population P with j th search agent constant by considering the upper bound ub and lower bound lb of moth-flame (mf) algorithm, the formula is as follow:

$$P_{mf} = lb_j + x_{i+1,j} \times \left(ub_j - lb_j\right), \tag{3}$$

where, $x_{i,j}$ is a random number between 0 and 1 when i is 1.

Fig. 3 is a comparison graphic of a two-dimensional diagram exhibiting the initialization method with and without tent chaotic mapping. It is easy to see that the starting points generated via tent chaos mapping initialization are more uniformly distributed throughout the search space. The y-axis represents the number of individuals in the sequence falling in the corresponding number of iterations on the x-axis.

The adaptive strategy that is being advocated takes advantage of DNA coding constraints and MFO evolution. We generalized the MFO algorithm with tent chaos as it can initialize the population uniformly. This present study utilized the moth (H) and flame (W) matrices, and fitness arrays (Hf) and (Wf) of these matrices from the original work [29].

$$H = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,d-1} & h_{1,d} \\ h_{2,1} & h_{2,2} & \cdots & \cdots & h_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ h_{n-1,1} & \cdots & \cdots & \cdots & h_{n-1,d} \\ h_{n,1} & h_{n,2} & \cdots & h_{n,d-1} & h_{n,d} \end{bmatrix}, Hf = \begin{bmatrix} Hf_1 \\ Hf_2 \\ \vdots \\ Hf_{n-1} \\ Hf_n \end{bmatrix} \tag{4}$$

$$W = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,d-1} & w_{1,d} \\ w_{2,1} & w_{2,2} & \cdots & \cdots & w_{2,d} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ w_{n-1,1} & \cdots & \cdots & \cdots & w_{n-1,d} \\ w_{n,1} & w_{n,2} & \cdots & w_{n,d-1} & w_{n,d} \end{bmatrix}, Wf = \begin{bmatrix} Wf_1 \\ Wf_2 \\ \vdots \\ Wf_{n-1} \\ Wf_n \end{bmatrix} \tag{5}$$

The detailed concept of the MFO is given in Ref. [29]. The author of MFO devised a spiral function called the logarithmic spiral function (see Eq. 3.12 of [29]) to produce a spiral path; however, we have customized that function for the population N in flame matrix W to update the position of H in Eq. (6):

$$y_i^{n+1} = \begin{cases} \beta_i \bullet x^{bt} \bullet \cos(2\pi t) + w_i, i \le NW \\ \beta_i \bullet x^{bt} \bullet \cos(2\pi t) + w_i, i \ge NW \end{cases} \tag{6}$$

where $\beta_i = \left|h_i^n - w_i\right|$, indicates the distance between the i-th moth h_i and its particular flame w_i, x is moth variance, b is a constant to identify the search space for helix shape within N population, and t is a random number between $(-1,+1)$, which shows the closeness of a particular h_i towards its w_i. As a h_i flies toward its w_i in a helix shape, t value will be discrete in only one-dimensional.

However, to improve the effectiveness of h_i and w_i in the first and last iteration (*iter*), and to achieve stable exploration and exploitation abil-

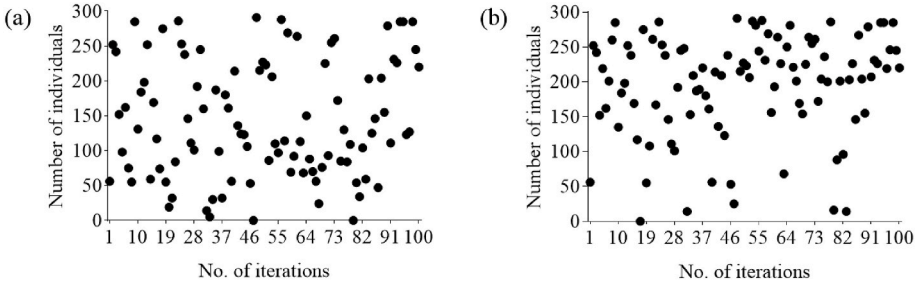

Fig. 3. Histogram distribution graph for population initialization (a) with and (b) without tent chaotic map with 100 iterations.

ities, we suggested an adaptable course to minimize the value of t for multi-dimension of global exploration and exploitation abilities in Eq. (7).

$$t = (\vartheta_1 - 1) \times r + 1. \tag{7}$$

$$\vartheta_1 = -1 + crnt_{iter}\left(\frac{-1}{max_{iter}}\right), \tag{8}$$

where ϑ_1 is a convergence constant that decreases from -1 to -2 to prove the global occurrence of exploration and exploitation in the current iteration ($crnt_{iter}$) with r interval, while, max_{iter} indicates the maximum numbers of *iter*.

Moths must modify their positions by switching to just one flame to avoid local minima. The fitness value is used to sort and modify the flames during each iteration. For instance, the first moth $h_{1,1}$ alter its position with its corresponding first flame $w_{1,1}$ and $h_{n,d}$ according to $w_{n,d}$. However, this positioning mechanism can affect the exploitation capability of the optimal solution of N-flame. To control this effect, we reduced the number of flames NW in each *iter* by constricting Eq. (9).

$$NW = round\left(NW_{L.iter} - crnt_{iter}\frac{(NW_{L.iter} - 1)}{max_{iter}}\right), \tag{9}$$

where $NW_{L.iter}$ is the flame number of the last iteration, which is minimized by the availability of the current flame ($crnt$) with *iter* value. It enables each moth's movement in the spiral with the corresponding flame.

Algorithm 2 differs from the MFO [29] by incorporating modified tent chaotic mapping for population initialization and customizing the logarithmic spiral function for better exploration and exploitation abilities, leading to improved convergence and global optima identification. The pseudocode of CMF is given in Algorithm 2, in which two initial populations P_{mf} are generated through the tent chaotic mapping and the inclusion of the flame W to work together to provide a balanced exploration and exploitation strategy. The tent chaotic mapping facilitates the exploration of a diverse solution space, while the inclusion of the flame acts as a starting point with known desirable qualities, guiding the optimization process. In the first iteration ($iter == 1$), at lines 15 and 16, the moths are sorted and updated based on their fitness values with respect to the flame fitness matrix (Wf). In subsequent iterations ($iter > 1$), at lines 18 and 19, the moths are again sorted based on the flame matrix at the last iteration ($W_{L.iter}$).

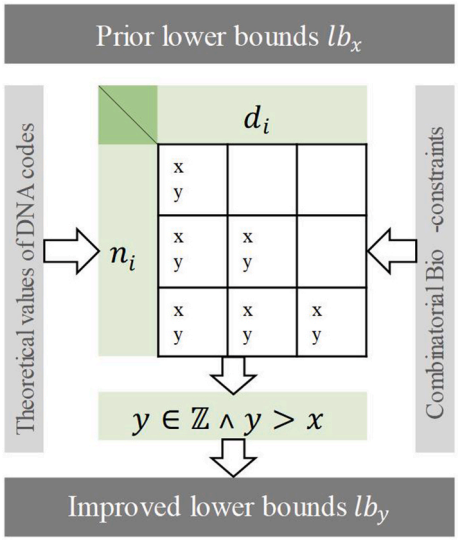

Fig. 4. Criteria of improved lower bounds with respect to prior lower bounds.

Algorithm 2. Pseudocode of CMF algorithm.

Algorithm 2. Pseudocode of CMF algorithm.

Input: Population size N for 2 solutions; moth H and flame W for i-th and j-th locations with f fitness function.

1: initialize population using Eq. (3) of tent chaotic map P_{mf};
2: **for** $i = 1{:}N$ **do**
3: **for** $j = 1{:}N$ **do**
4: calculate the random population $x_{i+1,j}$
5: **end for**
6: **end for**
7: **while** $iter < max_{iter}$
8: **if** $iter == 1$ **then**
9: add flame W to P_{mf} as an initial population
10: **else**
11: reduce NW using Eq. (9)
12: **end if**
13: Wf = fitness function of flame;
14: **if** $iter == 1$ **then**
15: sort moth w.r.t. Wf
16: update the moth using Eq. (6)
17: **else**
18: Sort moth w.r.t. $W_{L.iter}$
19: Update the moth using Eq. (6)
20: **end if**
21: reduce convergence constant ϑ_1;
22: **for** $i = 1{:}N$ **do**
23: **for** $j = 1{:}N$ **do**
24: calculate t and $\vartheta_1$using Eqs. (7) and (8)
25: update H position to specific W using Eq. (6)
26: **end for**
27: **end for**
28: **end while**

Output: Optimal solution with maximum fitness.

The evaluation of an algorithm's execution time is essential and is quantified through its computation time complexity. It relies on algorithm structure and input variables. Algorithm's execution breaks a set of numbers into halves, to search a particular field in its while loop. Therefore, it has a logarithmic time complexity, log(n); the running time of the algorithm is proportional to the number of times n because it divides the working area in half with each iteration. Meanwhile, the time complexity for the for loop is linear; the running time of the loop is directly proportional to n. As this algorithm use the logic of quick sorting. In contrast, if partitioning leads to almost equal subarrays, then the running time is the best, with time complexity as $\mathcal{O}$ (n*log n) that is a combination of logarithmic and linear time complexity. The overall computation time complexity can be expressed as follows:

$$\mathcal{O}(CMF) = \mathcal{O}(n * log\ n) \tag{10}$$

Compared to prior work [29] with the same purpose, the complexity for the best is $\mathcal{O}(n * log\ n)$. The decision to compare the time complexity of CMF with MFO is motivated by the practical use of MFO in this research. Although CMF has been modified from the original MFO, it still depends on MFO's essential functions and variables. Therefore, the comparison focuses exclusively on the time complexity of MFO. Thus, both CMF and MFO have the same time complexity.

The maximum iteration of the CMF algorithm determines the number of times the algorithm will search for an optimal solution before stopping. A higher maximum iteration count can increase the chances of finding an optimal and improved solution. In the context of DNA codes, the optimization algorithm is used to find better sequences that satisfy

certain constraints, also maximizing some objective function, such as coding capacity or stability. The criteria for improving lower bounds lb_y (Fig. 4) of the DNA codes refer to the minimum values that a function can achieve while still satisfying the constraints. In DNA coding and nanotechnology, the finding of DNA sequences that can store more information with higher reliability and stability and better resist mutations and nonspecific hybridization errors is improving the lower bounds. For this purpose, we derive new lower bounds in the following Theorems to strengthen the DNA sequences for unconstrained DNA strands to attain the desired inequality.

The fixed GC content δ and reverse complement (RC) constraints can be presented analogously as $N_4^{GC,RC}(n,d,\delta)$ for the $N_4(n,d,\delta)$ if all DNA sequences are with similar melting temperatures. Theorem 1 is based on lower bounds improvements by the objective function with the Hamming distance d, and the number of sequences n [31].

Theorem 1. *For the sequences* (x,y) *having the number of DNA codes* $n > 0$, *with* $0 \leq \delta \leq n$ *and* $0 \leq d \leq n$ *for the lower bound,*

$$N_4^{GC,RC}(n,d,\delta) = \begin{cases} 2 \text{ if } \delta = n/2 \\ 1 \text{ if } \delta \neq n/2 \end{cases} \tag{11}$$

Proof. If there are 2 codes, say $A^\delta T^\delta$ and $T^\delta A^\delta$ having GC content $\delta < \frac{n}{2}$ and RC (N_4^{RC}) there will be a next position where none of the words has A or T; thus, 1 of 2 codes should satisfy the Hamming distance and reverse constraints in that position. In contrast, for $\delta \neq n/2$, one word from $A^\delta T^\delta$ will satisfy both constraints as $T^\delta A^{n-\delta}$.

Apart from the lower bound's improvement, the consideration of RC constraints controls the errors created by the secondary structure [31]. For this, another mathematical model Theorem 2, is constructed to avoid the secondary structure (SS) from any sequence with lower free energy.

Theorem 2. DNA sequences x or y with 2n length in primer optimized codes are free from the secondary structure if the $d = H_d$, stem length > 1 and the free energy $E_{1,2n} \geq -2n$.

Proof. If any DNA sequence x or y has a secondary structure, it will have two disjoint sub-sequence as $x = y^{ss}$ with stem length l. The output will be contrapositive, i.e., if a sequence frees from a secondary complement sub-sequence, then the stem length will be more than 1. Meanwhile, a set of DNA codes with quinary alphabet {AG, AC, TC, CA, TT} of 2n length, the minimum free energy will avoid the secondary structure from the optimized DNA codes. For instance, from quinary codes, $E_{1,2n} \geq -5\left\lfloor \frac{2n}{2} \right\rfloor = -5n$, which has lower free energy that satisfies the RC constraints. Further details on free energy are referred to [32].

3.3. Bio-coding constraints

In DNA synthesis, errors, i.e., insertion, deletion, and substitution of bases, are prone to occur. It is estimated that 1% of DNA bases will contain errors during sequencing. It is easy to introduce nonspecific DNA hybridization during synthesis when there are some specific bases in the DNA sequence (for example, high GC content in the entire sequence and Homopolymers of particular bases). Hybridization reactions can adversely affect the normal sequencing process of DNA, leading to data read errors and failures because of sequencing deviations once they occur. As a result, it causes the unreliability of the DNA sequence [33]. It is critically considered for the sequence to conform with the bio-coding constraints and reduce nonspecific hybridization errors while reading and writing DNA sequences. This work has adopted the three prior bio-constraints, e.g., GC content [9], Homopolymer [21], and Hamming distance $d(x,y)$ [15], meanwhile, we have added another crucial constraints, RC.

3.3.1. RC constraints

The RC of a DNA sequence is the reverse of its complementary strand, where the complementary base pairs are formed by replacing A with T, C with G, and vice versa. For instance, A DNA sequence with length n will have a set of codes $(z_1 z_2 z_3 \ldots z_n)$, and the quaternary alphabet $z_i \in \{A, C, G, T\}$ indicates the four DNA bases that form a DNA sequence. A DNA sequence $z = z_1 z_2 z_3 \ldots z_n$ the reverse sequence $z^r = z_n z_{n-1} \ldots z_1$, complement sequence $z^c = z_1^c z_2^c z_3^c \ldots z_n^c$ and reverse-complement sequence $z^{rc} = z_n^c z_{n-1}^c \ldots z_1^c$. A c mark denotes the Watson-Crick complement of DNA nucleotide, thus $T^c = A, A^c = T, G^c = C$ and $C^c = G$ [34]. In real DNA sequences, AAGGTACT, AGTACCTT, and TGAAGCAT are the reverse, complement, and reverse-complement sequences for the TCATGGAA.

RC constraints are used in molecular biology for various purposes, i.e., searching for palindromic sequences, analyzing gene expression, genome assembly, and detecting mutations. It ensures accuracy and stability in DNA data storage by checking the accuracy of encoded DNA sequences and identifying errors. This constraint helps to maintain the stability of stored data by detecting degradation or mutations, which can introduce errors, and correcting them. These constraints relate to the secondary structure *SS* which refers to its 3D arrangement of the DNA molecule, which can generate errors in DNA synthesis by interfering with the accuracy of the process. During DNA synthesis, the double-stranded DNA molecule must unwind and separate to incorporate the complementary bases into the growing strand. If the DNA molecule has a complex *SS*, i.e., hairpin loops or stem-loop structures, it may be more difficult to unwind and separate the strands, leading to errors in the DNA synthesis process.

RC constraints eliminate *SS* errors by taking into account the base pairing properties of the DNA strands. By using these constraints, the design process considers both the sense and antisense strands of the DNA and ensures that complementary base pairs are not formed within the sequence, thereby avoiding the formation of unwanted *SS*. This leads to more reliable and accurate DNA sequences, free of structural errors that could affect their intended function.

3.3.2. Calculation of bio-constraints threshold

The above-mentioned four constraints (GC content, Homopolymer, $d(x,y)$, and RC constraints) can be calculated by the combinatorial mechanism, which combines these constraints in a weighted sum optimization problem to find the optimal DNA sequence that satisfies the threshold $f(x) =< T_m$. The objective function, $f(x) = \left(k/n\right) T_m$, can be calculated by the average (k) of the original sequence length (n) and the four constraints mentioned above.

4. Experiments and results

4.1. Experimental process

The experiments are conducted in an integrated environment with different tools. Firstly, benchmark medical image data of four cancers, including breast,[1] chest,[2] leukemia,[3] and skin,[4] are collected from various sources. This data is processed to convert into binary bits using a Python function (*TransBin*). Then, the binary data is mapped according to our novel encoding rules by importing Python packages (*codecs*) to generate the premier DNA sequences. We applied chaotic-based

[1] https://www.kaggle.com/datasets/shubamsumbria/breast-cancer-prediction.
[2] https://www.kaggle.com/datasets/mohamedhanyyy/chest-ctscan-images.
[3] https://www.kaggle.com/datasets/mehradaria/leukemia.
[4] https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000.

Table 4
BO-DNA's operators and parameter settings.

BO-DNA operators	Methods and parameters
Initial population	Tent chaotic map
Maximum iteration	Logarithmic spiral function
Population size	50
Parent selection	Random
Population selection	2 best individuals
Crossover probability	0.8
Crossover method	Arithmetic crossover
Mutation probability	0.05
GC content	40–60%

optimization algorithms to generate reliable DNA codes that satisfy the DNA coding constraints and avoid nonspecific hybridization errors. Tent chaotic map initialized the population for the moth-flame optimization algorithm, which enables our proposed biological encoding method to improve the lower bound by the maximum iteration supported by different coding constraints, i.e., GC content, Hamming distance, Homopolymer, and RC constraints. The sequences with improved lower bounds are computed with a bio-constraints threshold, which must satisfy the DNA coding constraints. The constrained sequences are considered optimal DNA sequences that can reliably store digital information. This information can also be decoded to retrieve the original information by utilizing our proposed decoding rule (Table 3).

All experiments are conducted in Windows 10 × 64 3.41 GHz Intel R-Core i7, 16 GB, and language Python with 3.8.11v. Three different state-of-the-art functions, unimodal, multimodal, and composite, have been employed to evaluate the performance of the chaotic-based moth-flame optimization algorithm. The theoretical details of these functions are in Ref. [29]. The parameter settings of this optimization algorithm are given in Table 4. As CMF is a heuristic algorithm that executes for n times, the average (AVG) and standard deviation (SD) metrics are used to assess the best solution of the algorithm. The lowest average score is the highest algorithm performance. The minimum standard deviation score is the maximum stability of the algorithm [29].

These metrics are used to compare the exploration and exploitation efficiencies of the proposed CMF with the existing algorithms. Additionally, the Friedman rank test [35] is used to demonstrate the statistical difference between our proposed algorithm and prior ones by using SOTA's functions. Moreover, the proposed CMF optimizer is customized for the maximum iteration to improve the lower bounds of DNA coding sets. We have delivered a mathematical formula to compute the bio-constraints threshold for the sequences with the improved lower bounds. Eventually, optimal DNA sequences generated by the proposed BO-DNA model are utilized to store the image dataset. We have also tested BO-DNA with other file formats, i.e., audio, text, and video dataset, and results are compared with prior studies.

4.2. Results

This section compares the CMF algorithm's performance efficiency with existing optimization algorithms. Then, BO-DNA encoding efficiencies are presented with different evaluations and indicators. The reasons that explain how these encoding results helped to generate a reliable DNA sequence are given in the Discussion and Limitation (Section 5).

4.2.1. SOTA functions

This study has utilized three different state-of-the-art (SOTA) functions to evaluate the performance of our proposed optimization algorithms. The details of these functions are given in Ref. [29]. However, we have selected those particular functions that do not perform optimally in our prior study [15,17,36]. Furthermore, we have compared performance with four existing optimization algorithms based on two metrics: average (AVG) and standard deviation (SD). Table 5 shows an improved trend of optimal performance in all functions for our optimization algorithm. For example, the SD of F16 is five times higher than the MFOS, while MFOL results have not been given (NA). Meanwhile, the optimal performance of the proposed CMF's functions has improved with a different magnitude than other studies [29,36]. The variances of functions, i.e., F6, F9, and F19, have pessimistic values, which can be helpful for the generation of reliable DNA codes. These functions emphasize exploitation (F1, F6), exploration (F9 and F13) performances, and a balanced optimal performance (F16, F19) among these SOTA functions.

4.2.2. Statistical test of SOTA-F

Friedman rank test provides an analysis that determines whether there is a statistically significant difference between means of three or more groups in which the same subjects appear in each [37]. As this study and our previous study [15] are based on optimization algorithms to construct DNA codes, taking into account we have tested a non-parametric statistical test, Friedman rank, on three groups of SOTA's functions (given in Table 5). Practically, an algorithm is considered statistically significant if the $P-value > 0.05$, which is presented as the threshold limit. The SOTA functions satisfying the threshold criteria are presented in Fig. 5. The line graph demonstrates the statistical significance of the BO-DNA algorithm compared to prior works MFO [29], MFOL [17], and MFOS [15] due to having more substantial and balanced exploration and exploitation abilities.

4.2.3. Lower bounds comparison

As retroactive studies, QRSS-MPA (d) [38], GCNSA (k) [39], and MFOS (m) [15] reported the complementary sequence forming secondary structure (SS) due to repeating subsequences and prone to errors in coding sets by reducing lower bounds. Therefore, this study evolved RC constraints with GC content, Hamming distance, and Homopolymer constraints $C^{GC,HP,RC}(n,w,d)$ to eliminate SS for reliable DNA storage; the higher the reliability, the lower the error probability. For improved lower bounds lb_y evaluation, we set $6 \leq n \leq 8$ and $3 \leq d < n$ inequalities bounds, and results are compared with prior lower bounds lb_x in Table 6 and Fig. 6. The selection reason for these specific bounds was to compare with prior studies QRSS-MPA (Table 4) [38], GCNSA (Table 10) [39], and MFOS (Table 4) [15].

In Table 6, a general trend indicates an adequate improvement (shown in bold entries) in the lower bounds of DNA codes compared to existing work. For instance, proposed work, BO-DNA (b) has 106% improved lower bounds than [39] in n = 7 and d = 5, while 0.10% reduced lower bounds than [15] in n = 8 and d = 4. Additionally, in a particular Hamming distance, e.g., d = 3, 15% of coding sets have been improved than [15]. In contrast, on a specific sequence length, e.g., n = 7, 32% of coding sets improved the lower bounds than [15]. Overall, an average lb improvements in all given bounds were 35%, 59%, and 12% improved than [38,39], and [15], respectively. Fig. 6 demonstrates that lb improvement is relatively less with a higher d, while it is considerable in higher n of the same constraints. Meanwhile, lb improvements directly favor enhancements of the DNA coding rates ($R = log_4 S/n$), S shows the number of coding sets, and n indicates the sequence length number [31]. For instance, the [39] provided $R = log_4 46/6 = 0.46$ when n = 6 and d = 3, while this work yielded a 0.54 coding rate when n = 6 and d = 3. In contrast with benchmark studies, the proposed work's coding rate was 0.04%, 0.14%, and 0.07% higher than [38,39], and [15], respectively.

4.2.4. BO-DNA encoding efficiency

The proposed model's efficiency is compared based on the density and reliability performance. Furthermore, we provided a comparison with prior benchmark studies.

4.2.4.1. Density and scalability analysis. We have encoded various files

Table 5
Comparison of different optimization algorithms' performances with BO-DNA by SOTA functions' values.

SOTA-F	Functions	Metrics	MFO [29]	FFA [36]	MFOL [17]	MFOS [15]	CMF
Unimodal functions	F1	AVG	8.63E+03	3.61E+03	1.49E+02	**1.06E+02**	**−3.17E+03**
		SD	1.48E+04	9.78E+03	3.80E+03	1.26E+03	**1.07E-02**
	F6	AVG	3.44E+01	5.04E+04	5.27E+01	**0.00E+00**	**−1.01E+00**
		SD	1.26E+04	5.10E+03	3.05E+03	**0.00E+00**	**−3.23E+04**
Multimodal functions	F9	AVG	1.92E+02	1.64E+02	2.42E+02	**0.00E+00**	**−2.44E+00**
		SD	6.75E+01	9.48E+01	5.93E+01	**0.00E+00**	**−3.70E+00**
	F13	AVG	1.00E+08	2.88E+07	**2.38E+04**	2.70E-03	**1.82E+11**
		SD	1.98E+08	**1.16E+08**	7.37E+08	5.45E-05	**1.01E+08**
Composite functions	F16	AVG	−1.02E+00	−1.02E+00	NA	−1.02E+00	**−6.13E+07**
		SD	7.51E-02	7.75E+02	NA	1.23E+01	**0.00E+00**
	F19	AVG	**−3.86E+00**	−3.85E+00	NA	−3.82E+00	**−8.02E+00**
		SD	3.40E-02	4.07E-02	NA	2.58E-02	**−4.92E+00**

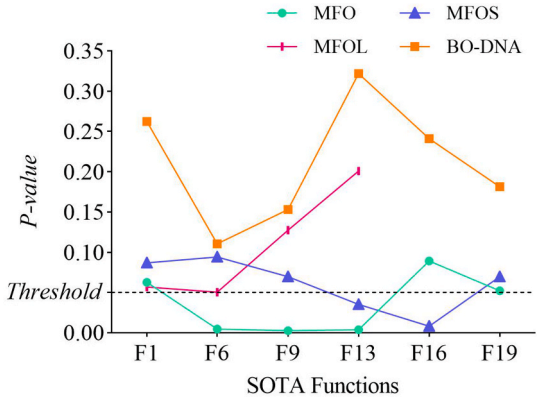

Fig. 5. Friedman rank test's comparison of BO-DNA with prior algorithms for SOTA functions.

Table 6
Lower bounds *lb* comparisons of BO-DNA (*b*) with QRSS-MPA (*q*) [38], GCNSA (*k*) [39], and MFOS (*m*) [15] using $C^{GC,NL,RC}(n, W, d)$.

n/d	*lb*	d = 3	d = 4	d = 5	d = 6	d = 7
6	lb_x	58^q, 46^k, 71^m	24^q, 25^k, 26^m	8^q, 8^k, 7^m	–	–
	lb_y	**90^b**	**29^b**	8^b	–	–
7	lb_x	124^q, 101^k, 134^m	45^q, 37^k, 63^m	16^q, 15^k, 23^m	**7^q**, 6^k, 5^m	–
	lb_y	**193^b**	**69^b**	**31^b**	6^b	–
8	lb_x	354^q, 299^k, 419^m	110^q, 94^k, **149^m**	36^q, 35^k, 51^m	15^q, **16^k**, 11^m	5^q, –, 6^m
	lb_y	**437^b**	133^b	**74^b**	11^b	6^b

with different sizes to assess the proposed encoding efficiencies. Table 7 reports the BO-DNA's performance of encodings. All files are 100% decoded with different densities, indicating slight variations for different files. The average density of cancer images is 1.7 bit/nt. Additionally, we have compared the BO-DNA's density and running time with the benchmark studies in Table 10 using the same image data.

Apart from image files, we have experimented with other file formats to assess the diverse efficiencies of BO-DNA encoding. However, why different files have different densities with the same encodings is a future question to be solved.

4.2.4.2. Reliability performance. To confirm the DNA code's reliabilities and satisfaction with biological combinatorial constraints, 500 random DNA sequences of image files are evaluated, providing the balanced GC content with BO-DNA. Fig. 7 compares unconstrained (a) and constrained (b) sequences without and with proposed BO-DNA encoding model, respectively. The constraint-based DNA sequences ensure the

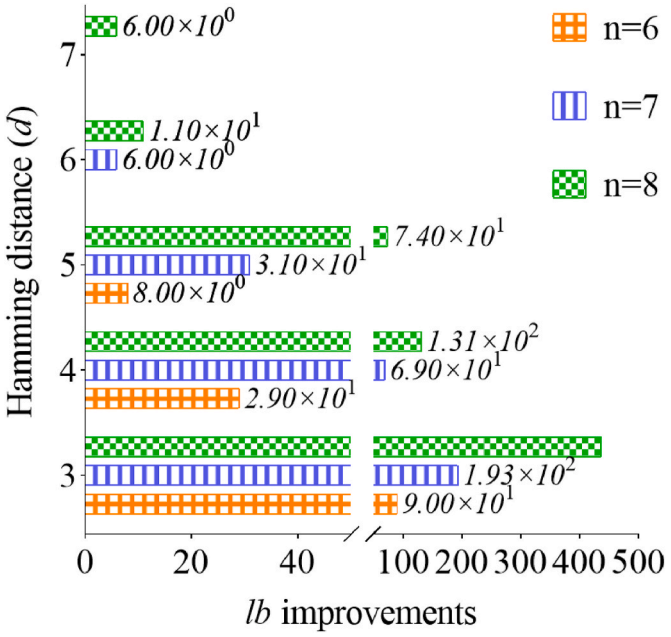

Fig. 6. Comparison of new lower bounds (lb_y) with different sequence lengths (n) based on d.

reduction of nonspecific hybridization errors, which cause DNA stabilities and reliability.

Additionally, the computation of the bio-constraints threshold by the objective function ($f(x)$) is presented in Table 8. It offers the optimal DNA sequence with the satisfied threshold of different files. For example, Fig. 8 exhibits a random sequence optimized and computed with different bio-coding to obtain constrained-based balanced sequences. The GC ratio and the number of bases of these constraints are evaluated for threshold satisfaction. The threshold results of the first optimal sequence from each DNA file are given in Table 8, to show the stable sequence. Although, the stability of DNA molecules depends on the chemical reaction, which varies with different factors, i.e., the interaction of hydrogen bonds between bases, melting temperature, and base—stacking connection between adjacent bases. As the purpose of our proposed optimized encoding for reliable DNA data storage, we only focused on designing an optimized coding method that must provide the constrained-based DNA sequences vital for a stable and reliable storage system. We have only considered the melting temperature (T_m) from the above factors to confirm the stability with constrained DNA sequences. The practical value of DNA concentration is adjusted to 200 nM, and salt concentration is 50 mM. The computed values of T_m are presented in Table 8 to measure the objective function $f(x)$ of bio-constraints threshold. This threshold is applied to new lower-bound DNA codes to

Table 7
The performance comparison of BO-DNA optimized encoding with different files on various factors.

File	Name.type	Size	Total DNA nucleotides	Time (s)	Density
Images	Breast cancer.png	3.67 MB	17056475	247.31	1.8
	Chest cancer.gif	8.63 MB	40015709	561.48	1.81
	Skin cancer.jpeg	7.22 MB	33539894	474.07	1.8
	Leukemia cancer.tiff	11.3 MB	55429634	790.74	1.72
Audio	Let Us Continue - Lyndon Baines Johnson.mp3	9.6 MB	45815244	638.19	1.81
Text	wiki_DNA_digital_data_storage.docx	28.7 KB	143460	2.00	1.7
Video	Albert Einstein Explains Theory of Relativity.mp4	5.36 MB	24972173	345.18	1.8

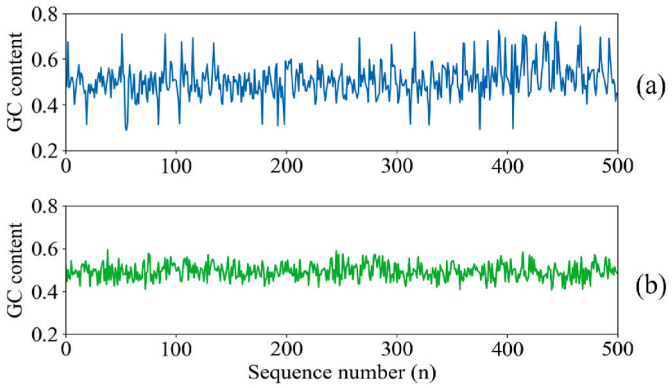

Fig. 7. The comparison of GC content satisfaction (a) without and (b) with proposed BO-DNA optimized encoding.

confirm the four bio-constraints, ensuring the optimal DNA sequence for reliable DNA storage. However, exploring the assessment of the DNA sequences of the proposed encoding with additional stability criteria mentioned earlier represents a promising path for future work.

4.2.4.3. Prior studies comparison. Meanwhile, the proposed BO-DNA encoding performance is compared with prior benchmark studies. Table 9 compares the proposed optimized encoding with an improved density and satisfying more coding constraints compared to many other studies. It signifies our proposed BO-DNA encoding to store various larger data files in smaller DNA nucleotides with improved densities and satisfying bio-constraints.

In Table 10, we present an additional comparison utilizing the same image data as in BO-DNA. This comparison evaluates the DNA storage density (D) and running time (T, in seconds) of our proposed BO-DNA approach, providing a fair scalability analysis with respect to prior work. The results demonstrate that BO-DNA achieves a higher storage density than existing methods, while its running time is slightly higher than many studies except [24]. The observed higher running time in our proposed BO-DNA approach is probably due to the implementation of the optimization technique on the proposed encoding map rules.

To further enhance the performance of BO-DNA, future research could explore more effective optimization techniques to mitigate the running time while maintaining or even improving storage density. Despite this potential area for improvement, it is essential to highlight that BO-DNA still offers several advantages over previous approaches. These advantages include enhanced density, which contributes to reduced synthesis and sequencing costs, minimized errors, and compliance with various bio-constraints, thus ensuring the reliability and practicality of the proposed method.

5. Discussion and limitation

The reliability and stability of current DNA storage development are still facing challenges and limitations in computer simulation and synthetic biology. In computer simulation, the construction of reliable DNA storage depends on the error-free and optimal generation of DNA nucleobases with maximum lower bounds [9]. Conversely, in synthetic biology, reliable DNA storage relies on chemical structure, storage environment, and sequencing [44]. As our focus is to design reliable DNA storage with computer simulation, we generated optimal DNA nucleobases by minimizing the nonspecific hybridization errors and maximizing the lower bounds to store huge data in a small amount of DNA without losing the original data. We propose a novel biologically optimized encoding model for DNA data storage (BO-DNA) to overcome the reliability problem. The BO-DNA approach introduces novel DNA mapping rules, and a significant contribution, alongside a customized optimization process of these rules. Additionally, the approach satisfies Bio-coding constraints, further contributing to its significance. The proposed BO-DNA model is a dependent system of mapping rules integrated with the optimization algorithm. To generate the constrained-based optimized DNA sequences, one has to rely on the proposed mapping method. Although this mapping method only generates the premier DNA nucleotides, which not satisfied the combinatorial coding constraints. However, the proposed rule-based mapping is highly scalable to map the binary data in DNA nucleotides with improved density. We customized the MFO algorithm with a tent chaotic map (CMF) as it can tackle difficult constraints and uncertain search space problems for various applications, including the sequence compression problem [29]. This optimization enables BO-DNA to improve the lower bounds, reducing nonspecific hybridization errors. We have constructed a criterion to judge the lower bounds improvement. The objective function computes the sequences satisfying this criterion to assess the reliability of the newly generated optimal DNA sequence.

In the experiments, we measured the optimization performance (Table 5) with state-of-the-art functions and found a balanced exploration and exploration capability of CMF. This capability is further evaluated (Fig. 5) with the Friedman rank test to find the statistical

Table 8
Evaluation of four bio-constraints on different files to compute the objective function for the optimal DNA sequence' reliability.

File name and type	n	H_d	Homopolymer	GC content	RC	T_m	$f(x)$
Breast cancer.png	107	104	3	49	91	78	45.01402
Chest cancer.gif	101	101	2	50	80	77	50.92673
Skin cancer.jpeg	107	106	3	54	96	80	54.72897
Leukemia cancer.tiff	111	109	3	51	78	78	49.47027
Let Us Continue - Lyndon Baines Johnson.mp3	107	107	2	55	84	80	53.08411
wiki_DNA_digital_data_storage.docx	106	104	4	48	89	77	50.99434
Albert Einstein Explains Theory of Relativity.mp4	104	100	3	50	71	77	48.56923

Fig. 8. Random sequence sample with balanced sequences satisfying the combinatorial coding constraints.

Table 9
The comparative analysis of proposed BO-DNA optimized encoding with prior studies.

Author	Year-Refs.	Coding method	Dataset	Sequence length	Density[a]	Bio-constraints
Church	2012-[4]	1 bit to 1 base	English text, JPG images, computer code	115	0.83	Homopolymer
Goldman	2013-[6]	Rotating encoding	Text file, JPEG file, MP3 file	117	0.33	Run length >2
Grass	2015-[33]	Reed–Solomon (RS) coding	Text from the Swiss Federal Charter	158	1.14	No-runlenght
Blawat	2016-[20]	6 bis to 3 bases	MPEG compressed movie sequence	230	1.08	Hetero-dimerization, Hairpin
Bornholt	2016-[40]	Rotating encoding	Three JPG files	120–150	0.58	Homopolymer
Erlich	2017-[5]	DNA fountain	Text file, SVG file, Video file	152	1.57	GC content, no-runlength
Organick	2018-[7]	RS coding	high-definition video, images	150–200	1.1	GC content, no-runlength
Choi	2020-[41]	One character	Text file	85	3.37	N/A
Yazdi	2020-[19]	14 bits to 8 bases	Two JPEG images	800–1000	1.74	Homopolymer, GC content
Jeong	2021-[42]	DNA fountain encoding	JPG file	152	1.53	Homopolymer-run length
Cao	2022-[27]	Fountain encoding	JIFI, MP4	162	1.41	GC content, Homopolymer
Song	2022-[43]	De Bruijn graph and greedy path search	Random 6.8 MB	190	1.30	GC content, Melting temperature
Rasool	2023	Biologically optimized encoding (BO-DNA)	Medical images (PNG, GIF, JPEG, TIFF) and MP3, TXT, MP4 files	120	**1.77**	GC content, Hamming distance, RC, Homopolymer

[a] bit/nucleotides without primer.

significance of the CMF algorithm. These results strengthen our idea for the practical implementation of an optimization algorithm for DNA code generation with adequate performance. The improved lower bounds (12%–59%) of DNA coding sets are received and compared with prior studies [38,39], and [15], presented in Fig. 6 and Table 6. There are few studies on improving lower bounds that we have not considered in our comparisons due to using different optimization algorithms and DNA coding constraints. Eventually, BO-DNA encoding efficiencies are presented and compared with various evaluations (Figs. 7 and 8, and Tables 7–10). The encoding efficiencies significantly improved in many aspects. For instance, BO-DNA model reported 1.77 average density, which is higher to a different extent than existing studies, i.e., 0.40% higher than Song's work published in 2022 [43], and 1.44% higher than Goldman's reported in 2013 [6]. Meanwhile, the encoding time complexity is also adequate, i.e., a medical image of 11.3 MB is encoded and decoded successfully in 13 min. It must be noted that the given scalability analysis is also proportional to the computer specifications. For the reliability analysis, we analyzed random sequence samples to confirm the constraint satisfaction issue. For example, Fig. 7(b) demonstrates the GC content constraint within the 40%–60% ratio, which is the benchmark. Furthermore, the remaining coding satisfaction results are reported in Table 8. For example, we set the limitation ≤ 3 Homopolymers were allowed, and resultant sequences satisfied this constraint in many files except the. docx file, which is still not alarming for error generation. Additionally, to confirm the reliability of newly generated DNA sequence, the results of the objective function satisfy the criteria $f(x) \leq T_m$ that shows the new sequence can be synthesized and sequenced without propagating the nonspecific hybridization errors. This threshold is achieved due to the satisfaction of four constraints obtained due to the implementation of optimization technique with the

Table 10
The comparison of two parameters (P); storage density (D) and running time (T) in seconds between BO-DNA and prior studies using the same image data as BO-DNA.

Authors [Ref.]	P	Image files			
		PNG	GIF	JPEG	TIFF
Church [4]	D	1	1	1	0.97
	T	209.33	445.93	454.68	617.76
Goldman [6]	D	1.3	1.3	1.3	1.3
	T	187.48	499.117	401.007	655.72
Erlich [5]	D	1.5	1.5	1.6	1.3
	T	221.3	430.545	361.57	554.13
Yazdi [19]	D	1.57	1.55	1.6	1.52
	T	204.7	408.92	387.29	571.84
Yin-Yang [24]	D	1.7	1.7	1.78	1.76
	T	3893.4	8643.3	6204.5	5304.6
Rasool (BO-DNA)	D	1.8	1.81	1.8	1.72
	T	247.31	561.48	474.07	790.74

rule-based mapping for the data storage in DNA. Eventually, it was reported that the proposed work outperformed the prior work. It must also be noted that this work provided the optimal DNA sequences which satisfy the four different coding constraints, while previously, often studies generated sequences with 2 or 3 coding constraints.

The results demonstrated the improved lower bounds and encoding efficiency with high density. However, let us ask a more modest question. How can the improvement in lower bounds of DNA sequence data enable the achievement of reliable DNA data storage? A precise answer negates accomplishing all objectives except the stable DNA codes by reducing the nonspecific hybridization errors. Molecular-based computation delivers a set of sequences with the corrupt version of DNA codes, including symbol deletion or substitution and high-magnitude errors in the codewords, which lose the sequences during the hybridization process [45,46]. These massive errors decrease the lower bounds between the different numbers of sequences. Consequently, the DNA synthesis process becomes noisy. Then, it is more expensive to synthesize and relatively reduces the stability of DNA data storage, which can be biotechnologically quantified, controlled, and engineered [47]. Thus, improving lower bounds with optimization studies can significantly affect the reeducation of nonspecific hybridization errors.

Despite the significant improvements and efficiencies of the BO-DNA optimized encoding model for DNA data storage, various limitations and gaps have been observed based on our computer simulations. BO-DNA model provides practical evidence of optimization implications for DNA sequence encoding; in the future, it can be applied to the prior benchmark DNA storage studies, i.e., Fountain codes [5] and Yin-Yang codes [24]. Currently, this study focuses on constructing optimized and constrained-based DNA sequences; in the future, however, error correction mechanisms, i.e., Reed–Solomon (RS) codes [42], can be implemented to control the insertion, deletion, and substitution errors. It must be noted this study tackles the nonspecific hybridization errors which often cause the stability of DNA sequence and storage. One can propose a new rule-based mapping method on 2-base, 3-base, and 4-base encodings to produce more reliable sequences with other DNA coding constraints, e.g., edit distance and forbidden coding constraints [48]. Meanwhile, other SOTA optimization algorithms [36] can also be tested to assess the DNA sequence's convergence abilities and new lower bounds.

6. Conclusion

The reliability of stored data is adversely affected by nonspecific hybridization errors, which frequently occur during DNA synthesis and sequencing. This study proposes a novel biologically optimized encoding model for DNA data storage (BO-DNA). The purpose of this study is to generate optimal DNA sequences to tackle the reliability problem which occurs due to nonspecific hybridization errors. An innovative rule-based mapping method is developed to avoid data loss during binary encoding. Meanwhile, minimizing nonspecific hybridization errors is achieved by applying a customized optimization algorithm (CMF) based on a tent chaotic map. Newly generated optimal DNA sequences are tested for robustness by four bio-constraints. The experiments are processed on medical data of four cancer images to test the BO-DNA model's efficiency and compare it with other files and benchmark studies. The results of the CMF algorithm with SOTA functions (Table 5) and statistical tests (Fig. 5) revealed the balanced exploitation and exploration abilities compared to prior studies. It demonstrates the optimization efficiencies to generate DNA sequences with 12%–59% improved lower bounds (Table 6). It means that DNA codes in a particular sequence have more capacity to store the data, improving the density. Furthermore, the yielded sequences are computed with four bio-constraints thresholds (Table 8) to determine the sequence reliability. The optimal DNA sequences are obtained based on the objective function satisfying the coding constraints. The constraint satisfaction concludes the elimination of nonspecific hybridization errors and validates the stability of synthesized DNA. Eventually, the BO-DNA encoding model is compared with various files, emphasizing the widely adopted efficiency of storing different files. Additionally, the proposed work is compared with prior studies to present the significance of optimizing encoding for data storage. It concludes the practical implication of an optimization algorithm for DNA data storage by storing larger files in shorter DNA sequences, improving density from 0.40% to 1.44%, achieving optimal sequences, and reducing nonspecific hybridization errors, ensuring the reliability of the DNA storage system. Apart from the vital performance, the proposed BO-DNA encoding has limitations and future gaps, discussed in Section 5.

Data availability

The data and codes for this work are available at https://github.com/abdul-rasool/BO-DNA.

Funding

This work is supported by the National Key Research and Development Program of China under fund numbers 2020YFA0909100, 2021YFF1200100, and 2021YFF1200104.

Declaration of competing interest

The authors declare no conflict of interest.

Acknowledgments

The authors would like to thank all the anonymous reviewers for their insightful comments and constructive suggestions that have obviously upgraded the quality of this manuscript.

References

[1] L. Organick, et al., Probing the physical limits of reliable DNA data retrieval, Nat. Commun. 11 (1) (2020) 616, https://doi.org/10.1038/s41467-020-14319-8, 2020/01/30.
[2] K. Matange, J.M. Tuck, A.J. Keung, DNA stability: a central design consideration for DNA data storage systems, Nat. Commun. 12 (1) (2021) 1358, https://doi.org/10.1038/s41467-021-21587-5, 2021/03/01.
[3] Y. Dong, F. Sun, Z. Ping, Q. Ouyang, L. Qian, DNA storage: research landscape and future prospects, Natl. Sci. Rev. 7 (6) (2020) 1092–1107, https://doi.org/10.1093/nsr/nwaa007.
[4] G.M. Church, Y. Gao, S. Kosuri, Next-generation digital information storage in DNA, Science 337 (6102) (Sep 28 2012), https://doi.org/10.1126/science.1226355, 1628-1628.
[5] Y. Erlich, D. Zielinski, DNA Fountain enables a robust and efficient storage architecture, Science 355 (6328) (Mar 3 2017) 950–953, https://doi.org/10.1126/science.aaj2038.

[6] N. Goldman, et al., Towards practical, high-capacity, low-maintenance information storage in synthesized DNA, Nature 494 (7435) (Feb 7 2013) 77–80, https://doi.org/10.1038/nature11875.
[7] L. Organick, et al., Random access in large-scale DNA data storage, Nat. Biotechnol. 36 (3) (2018/03/01 2018) 242–248, https://doi.org/10.1038/nbt.4079.
[8] X. Li, Z. Wei, B. Wang, T. Song, Stable DNA sequence over close-ending and pairing sequences constraint (in English), Frontiers in Genetics, Original Research 12 (2021), https://doi.org/10.3389/fgene.2021.644484, 2021-May-17.
[9] B. Cao, X. Li, X.K. Zhang, B. Wang, Q. Zhang, X.P. Wei, Designing uncorrelated address constrain for DNA storage by DMVO algorithm, IEEE ACM Trans. Comput. Biol. Bioinf 19 (2) (Mar-Apr 2022) 866–877, https://doi.org/10.1109/tcbb.2020.3011582.
[10] M. Welzel, et al., DNA-Aeon provides flexible arithmetic coding for constraint adherence and error correction in DNA storage, Nat. Commun. 14 (1) (2023/02/06 2023) 628, https://doi.org/10.1038/s41467-023-36297-3.
[11] K. Makarychev, M.Z. Racs, C. Rashtchian, S. Yekhanin, Batch optimization for DNA synthesis, IEEE Trans. Inf. Theor. 68 (11) (Nov 2022) 7454–7470, https://doi.org/10.1109/tit.2022.3184903.
[12] X.Y. Wang, Y.P. Li, Chaotic image encryption algorithm based on hybrid multi-objective particle swarm optimization and DNA sequence, Opt Laser. Eng. 137 (Feb 2021), 106393, https://doi.org/10.1016/j.optlaseng.2020.106393.
[13] Z.X. Zhu, J.R. Zhou, Z. Ji, Y.H. Shi, DNA sequence compression using adaptive particle swarm optimization-based memetic algorithm, IEEE Trans. Evol. Comput. 15 (5) (Oct 2011) 643–658, https://doi.org/10.1109/tevc.2011.2160399.
[14] B. Cao, X.K. Zhang, J.Q. Wu, B. Wang, Q. Zhang, X.P. Wei, Minimum free energy coding for DNA storage, IEEE Trans. NanoBioscience 20 (2) (Apr 2021) 212–222, https://doi.org/10.1109/tnb.2021.3056351.
[15] A. Rasool, Q. Jiang, Y. Wang, X. Huang, Q. Qu, J. Dai, Evolutionary approach to construct robust codes for DNA-based data storage (in English), Frontiers in Genetics, Original Research 14 (2023), https://doi.org/10.3389/fgene.2023.1158337, 2023-March-20.
[16] A. Doricchi, et al., Emerging approaches to DNA data storage: challenges and prospects, ACS Nano 16 (11) (2022) 17552–17571, https://doi.org/10.1021/acsnano.2c06748, 2022/11/22.
[17] A. Rasool, Q. Qu, Q. Jiang, Y. Wang, A strategy-based optimization algorithm to design codes for DNA data storage system, in: Algorithms and Architectures for Parallel Processing, Springer International Publishing, 2022, pp. 284–299. Cham.
[18] J. Davis, Microvenus, Art J. 55 (1) (1996) 70–74, https://doi.org/10.1080/00043249.1996.10791743, 1996/03/01.
[19] S.M.H.T. Yazdi, R. Gabrys, O. Milenkovic, Author correction: portable and error-free DNA-based data storage, Sci. Rep. 10 (1) (2020) 7026, https://doi.org/10.1038/s41598-020-60080-9, 2020/04/22.
[20] M. Blawat, et al., Forward error correction for DNA data storage, Procedia Comput. Sci. 80 (2016) 1011–1022, https://doi.org/10.1016/j.procs.2016.05.398, 2016/01/01/.
[21] W.H. Press, J.A. Hawkins, S.K. Jones, J.M. Schaub, I.J. Finkelstein, HEDGES error-correcting code for DNA storage corrects indels and allows sequence constraints, Proc. Natl. Acad. Sci. USA 117 (31) (2020) 18489–18496, https://doi.org/10.1073/pnas.2004821117.
[22] P.M. Schwarz, B. Freisleben, NOREC4DNA: using near-optimal rateless erasure codes for DNA storage, BMC Bioinf. 22 (1) (2021) 406, https://doi.org/10.1186/s12859-021-04318-x, 2021/08/17.
[23] P. Mishra, C. Bhaya, A.K. Pal, A.K. Singh, Compressed DNA coding using minimum variance huffman tree, IEEE Commun. Lett. 24 (8) (Aug 2020) 1602–1606, https://doi.org/10.1109/lcomm.2020.2991461.
[24] Z. Ping, et al., Towards practical and robust DNA-based data archiving using the yin–yang codec system, Nature Computational Science 2 (4) (2022) 234–242, https://doi.org/10.1038/s43588-022-00231-2, 2022/04/01.
[25] B. Cao, P.J. Shi, Y.F. Zheng, Q. Zhang, FMG: an observable DNA storage coding method based on frequency matrix game graphs, Comput. Biol. Med. 151 (Dec 2022), 106269, https://doi.org/10.1016/j.compbiomed.2022.106269.
[26] W.G. Chen, et al., An artificial chromosome for data storage, Natl. Sci. Rev. 8 (5) (May 2021), nwab028, https://doi.org/10.1093/nsr/nwab028.
[27] B. Cao, X. Zhang, S. Cui, Q. Zhang, Adaptive coding for DNA storage with high storage density and low coverage, npj Systems Biology and Applications 8 (1) (2022) 23, https://doi.org/10.1038/s41540-022-00233-w, 2022/07/04.
[28] G. Kaur, S. Arora, Chaotic whale optimization algorithm, Journal of Computational Design and Engineering 5 (3) (2018) 275–284, https://doi.org/10.1016/j.jcde.2017.12.006.
[29] S. Mirjalili, Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm, Knowl. Base Syst. 89 (Nov 2015) 228–249, https://doi.org/10.1016/j.knosys.2015.07.006.
[30] C. Wen, et al., Modified remora optimization algorithm with multistrategies for global optimization problem, 3604, Mathematics 10 (19) (2022) [Online]. Available: https://www.mdpi.com/2227-7390/10/19/3604.
[31] A. Rasool, Q. Qu, Y. Wang, Q.S. Jiang, Bio-constrained codes with neural network for density-based DNA data storage, Mathematics 10 (5) (Mar 2022), https://doi.org/10.3390/math10050845, 845.
[32] M.S. Adams, B.M. Znosko, Thermodynamic characterization and nearest neighbor parameters for RNA duplexes under molecular crowding conditions, Nucleic Acids Res. 47 (7) (2019) 3658–3666, https://doi.org/10.1093/nar/gkz019.
[33] R.N. Grass, R. Heckel, M. Puddu, D. Paunescu, W.J. Stark, Robust chemical preservation of digital information on DNA in silica with error-correcting codes, Angew Chem. Int. Ed. Engl. 54 (8) (Feb 16 2015) 2552–2555, https://doi.org/10.1002/anie.201411378 (in eng).
[34] N. Aboluion, D.H. Smith, S. Perkins, Linear and nonlinear constructions of DNA codes with Hamming distance d, constant GC-content and a reverse-complement constraint, Discrete Math. 312 (5) (2012) 1062–1075, https://doi.org/10.1016/j.disc.2011.11.021, 2012/03/06/.
[35] R. Eisinga, T. Heskes, B. Pelzer, M. Te Grotenhuis, Exact p-values for pairwise comparison of Friedman rank sums, with application to comparing classifiers, BMC Bioinf. 18 (1) (2017) 68, https://doi.org/10.1186/s12859-017-1486-2, 2017/01/25.
[36] E. Emary, H.M. Zawbaa, K.K.A. Ghany, A.E. Hassanien, B. Parv, Firefly optimization algorithm for feature selection, in: Presented at the Proceedings of the 7th Balkan Conference on Informatics Conference, Craiova, Romania, 2015, https://doi.org/10.1145/2801081.2801091 [Online]. Available:.
[37] D. Berrar, Using p-values for the comparison of classifiers: pitfalls and alternatives, Data Min. Knowl. Discov. 36 (3) (2022) 1102–1139, https://doi.org/10.1007/s10618-022-00828-1, 2022/05/01.
[38] Q. Yin, Y. Zheng, B. Wang, Q. Zhang, Design of constraint coding sets for archive DNA storage, IEEE ACM Trans. Comput. Biol. Bioinf 19 (6) (2022) 3384–3394, https://doi.org/10.1109/TCBB.2021.3127271.
[39] B. Cao, B. Wang, Q. Zhang, GCNSA: DNA storage encoding with a graph convolutional network and self-attention, iScience 26 (3) (2023), 106231, https://doi.org/10.1016/j.isci.2023.106231, 2023/03/17/.
[40] J. Bornholt, R. Lopez, D.M. Carmean, L. Ceze, G. Seelig, K. Strauss, Toward a DNA-based archival storage system, IEEE Micro 37 (3) (2017) 98–104, https://doi.org/10.1109/MM.2017.70.
[41] Y. Choi, et al., DNA micro-disks for the management of DNA-based data storage with index and write-once-read-many (WORM) memory features, Adv. Mater. 32 (37) (Sep 2020), 2001249, https://doi.org/10.1002/adma.202001249.
[42] J. Jeong, et al., Cooperative sequence clustering and decoding for DNA storage system with fountain codes, Bioinformatics 37 (19) (2021) 3136–3143, https://doi.org/10.1093/bioinformatics/btab246.
[43] L. Song, et al., Robust data storage in DNA by de Bruijn graph-based de novo strand assembly, Nat. Commun. 13 (1) (2022) 5361, https://doi.org/10.1038/s41467-022-33046-w, 2022/09/12.
[44] A. Baoutina, S. Bhat, L. Partis, K.R. Emslie, Storage stability of solutions of DNA standards, Anal. Chem. 91 (19) (Oct 2019) 12268–12274, https://doi.org/10.1021/acs.analchem.9b02334.
[45] A.K.-Y. Yim, et al., The essential component in DNA-based information storage system: robust error-tolerating module, 49, Front. Bioeng. Biotechnol. 2 (2014), https://doi.org/10.3389/fbioe.2014.00049, 2014.
[46] M. Li, et al., A self-contained and self-explanatory DNA storage system, Sci. Rep. 11 (1) (2021), 18063, https://doi.org/10.1038/s41598-021-97570-3, 2021/09/10.
[47] L.F. Song, Z.H. Deng, Z.Y. Gong, L.L. Li, B.Z. Li, Large-Scale de novo Oligonucleotide Synthesis for Whole-Genome Synthesis and Data Storage: challenges and Opportunities, Front. Bioeng. Biotechnol. 9 (Jun 2021), 689797, https://doi.org/10.3389/fbioe.2021.689797.
[48] S. Lebre, O. Gascuel, The combinatorics of overlapping genes, J. Theor. Biol. 415 (Feb 2017) 90–101, https://doi.org/10.1016/j.jtbi.2016.09.018.