ISO/IEC JTC 1/SC 29/WG 1
(ITU-T SG16)

Coding of Still Pictures

JBIG	JPEG
Joint Bi-level Image Experts Group	Joint Photographic Experts Group

TITLE:	**DNA-based Media Storage: State-of-the-Art, Challenges, Use Cases and Requirements version 7.0**
SOURCE:	Contributors: Marc Antonini (editor), Luis Cruz, Eduardo da Silva, Melpomeni Dimopoulou, Touradj Ebrahimi (editor), Siegfried Foessel, Eva Gil San Antonio, Gloria Menegaz, Fernando Pereira (editor), Xavier Pic, António Pinheiro, Mohamad Raad
PROJECT:	JPEG DNA Exploration
STATUS:	**Final**
REQUESTED ACTION:	For information and feedback
DISTRIBUTION:	Public

Contact:
ISO/IEC JTC 1/SC 29/WG 1 Convener – Prof. Touradj Ebrahimi
EPFL/STI/IEL/GR-EB, Station 11, CH-1015 Lausanne, Switzerland
Tel: +41 21 693 2606, Fax: +41 21 693 7600, E-mail: Touradj.Ebrahimi@epfl.ch

DNA-based Media Storage

State-of-the-Art, Challenges, Use Cases and Requirements

v 7.0

Executive Summary

DNA is a macromolecule that is essential for any form of life and is made of simple units that line up in a particular order within this large molecule. The order of these units usually carries the genetic information for a specific life organism, similar to how the order of letters in a text carries information. In practice, this means that DNA molecules may be artificially created with specific unit orders, notably to store some relevant sequence of information.

In digital media information, notably images, the relevant representation symbols, e.g. quantized DCT coefficients, are expressed in bits (binary units) but they could be expressed in any other units, for example, the DNA units which follow a quaternary (4-ary) representation basis. This would mean that artificial DNA molecules may be created with a specific DNA configuration which stores some media representation symbols or bits, e.g. the symbols or bits of a JPEG compressed image, thus leading to DNA-based media storage as a form of molecular data storage.

To make it more interesting, the DNA data storage density seems to be extremely high, notably beyond any available storage technology, but also energy-friendly and very durable. In this context, DNA storage implies DNA synthesis/storage and DNA sequencing/access, which are currently rather complex and expensive processes but should become increasingly affordable in the coming years.

This exciting story directly leads to the purpose of this document, which is basically to review and discuss:

1. DNA-based media storage basics, architectures and technology state-of-the-art
2. DNA-based media storage challenges
3. DNA-based media storage use cases and requirements
4. Main players in DNA-based media storage
5. JPEG role and next steps in JPEG DNA-based media storage activity

As a minimum, the JPEG committee can launch an activity to convert its existing image coding formats from compressed binary representation to compressed DNA 4-ary representation, according to appropriate requirements. Standardized image coding solutions along with complementary appropriate tools, such as error resiliency and associated metadata, which particularly suit the requirements of DNA information storage, are good directions for JPEG to explore.

The scope of JPEG DNA is the creation of a standard for efficient coding of images that considers

biochemical constraints and offers robustness to noise introduced by the different stages of the storage process that is based on DNA synthetic polymers.

1. Background and Motivation

DNA is a macromolecule that is essential for any form of life and is made of simple units that line up in a particular order within this large molecule. The order of these units usually carries the genetic information for a specific life organism, similar to how the order of letters in a text carries information. However, it is also possible to create artificial DNA molecules with specific DNA unit orders, notably to store some relevant sequence of information.

In digital media information, notably images, the relevant representation symbols, e.g. quantized DCT coefficients, are expressed in bits (binary units) but they could be expressed in any other units, for example, the DNA units which follow a 4-ary representation basis. This would mean that artificial DNA molecules may be created with a specific DNA unit configuration which stores some media representation symbols or bits, e.g. the symbols or bits of a JPEG compressed image, thus leading to DNA-based media storage as a form of molecular data storage. In this context, DNA storage implies DNA synthesis/storage and DNA sequencing/access, which are currently rather complex and expensive processes However these processes are expected to become increasingly affordable in the coming years.

To make this storage mechanism more interesting, the DNA data storage density seems to be extremely high, notably beyond any available storage technology. Moreover, DNA-based storage is also extremely stable, as demonstrated by the complete genome sequencing of a fossil horse that lived 700,000 years ago [1]. And, even more interesting, storing DNA does not require much energy. On the other hand, current magnetic and optical data-storage systems cannot last for more than a century and they spend large amounts of energy. In summary, DNA-based storage may be a very powerful alternative to the current data-storage solutions, which seem to have rather serious limitations, notably in terms of storage capacity, duration, and energy consumption.

This exciting story directly leads to the motivation of this document, which is basically to review and discuss:

1. DNA-based media storage basics, architectures, and technology state-of-the-art
2. DNA-based media storage challenges
3. DNA-based media storage use cases and requirements
4. Main players in DNA-based media storage
5. JPEG role and next steps in JPEG DNA-based media storage activity

2. JPEG Standards for Storage and Archival

JPEG standards have been used in the storage and archival of digital pictures as well as moving images. The most popular format for storage and archival of digital pictures is the popular legacy JPEG format as described in ISO/IEC 10918 and, in particular, in parts 1, 3, and 5 of the latter standard.

While the legacy JPEG format is widely used for photo storage in SD cards, as well as archival of pictures by consumers, JPEG 2000 as described in ISO/IEC 15444 is used in many archival applications, notably for the preservation of cultural heritage in the form of visual data as pictures and video in digital format. Notable examples are the Library of Congress, Library and Archives Canada, Chronicling America website, and the Google Library Project. Because of its use in digital cinema, JPEG 2000 is also used for archival of movies in digital form.

In terms of technology, both legacy JPEG and JPEG 2000 formats are based on a transform-quantization-entropy coding pipeline with JPEG using the Discrete Cosine Transform (DCT) and JPEG 2000 using the Discrete Wavelet Transform (DWT), followed by quantization, coefficient reordering, and entropy coding. The legacy JPEG format has been extended to define JPEG XT, as described in ISO/IEC 18477, to include features attractive for archival applications such as lossless coding, while being backward compatible with the popular legacy JPEG format.

The latest JPEG image coding format called JPEG XL, as described in ISO/IEC 18181, also offers a number of attractive features important to archival applications, such as lossless compression and lossless transcoding from legacy JPEG to JPEG XL, resulting in smaller file sizes without numerical loss in the pixel values.

3. DNA-based Media Storage Technologies

Deoxyribonucleic acid (DNA) is a molecule composed by two polynucleotide chains that coil around each other to form a double helix, carrying genetic instructions for the development, functioning, growth, and reproduction of all known organisms and many viruses. DNA and ribonucleic acid (RNA) are nucleic acids. Alongside proteins, lipids, and complex carbohydrates (polysaccharides), nucleic acids are one of the four major types of macromolecules that are essential for all known forms of life [2].

A so-called *base* or *nucleotide* (nt) is a unit of the DNA molecule. There are four different DNA bases: Adenine (A) and Guanine (G) are the larger purines; Cytosine (C) and Thymine (T) are the smaller pyrimidines, see Figure 1. The sequence of bases (for example, CAG) in a specific DNA represents the genetic code.

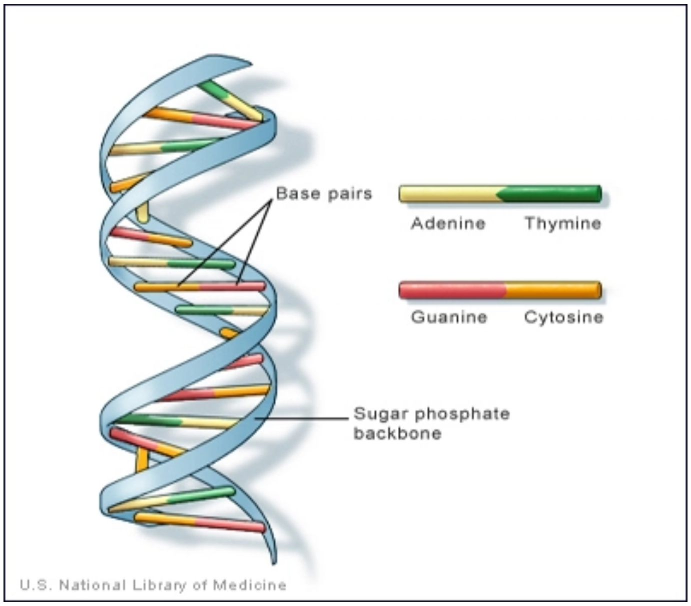


Figure 1 - DNA molecule and its units/nucleotides [3].

DNA fragments (i.e. sequences of the A, G, C, and T units) as codestream embodiments for stored data may be synthesized/written/printed onto a DNA microplate (i.e. a flat plate with multiple "wells" used as small test tubes) or kept in a test tube and stored somewhere cool, dark, and dry, such as a refrigerator.

Recovering/reading the DNA stored information involves rehydrating the sample, amplifying the fragments using PCR (Polymerase Chain Reaction), and then sequencing and reassembling the full nucleotide code. Provided the user knows the strategy employed to generate the DNA code, they can then decode the original message.

In practice, the mechanism that Nature has been using to store the information of life may be now used to store any other type of information. The objective is to store information in synthetic DNA molecules created in a lab, naturally not DNA from humans or other living things. Just as with other storage systems, the data can be encrypted before it is written to the storage medium [4].

Moreover, this biological mechanism has the very appealing feature of reaching spectacular data storage densities, much beyond the current electronics mechanisms. According to [1], *"..., all the world's current storage needs for a year could be well met by a cube of DNA measuring about one meter on a side."*

However, there are several challenges to overcome to successfully store media information using DNA [5]. The DNA-coded information stream must respect some biological constraints on the combinations of the A, G, C, T bases that form a DNA fragment to reduce the synthesis and sequencing errors. There is also a need to overcome “biological errors”, mainly substitutions or indels (insertions or deletions of (a)_nucleotide(s)), when storing and recovering information in DNA [6], that is, DNA needs to be viewed as a naturally noisy channel for which appropriately resilient codes need to be defined.

3.1 DNA-based Media Storage: End-to-end Architecture

The overall workflow of an end-to-end DNA-based media storage architecture includes the following phases:

1. **Encoding** - This phase corresponds to the conversion of the visual information into a DNA representation composed of molecules made of sequences of A, G, T, and C; the visual information may be available in multiple representation forms and encoding may happen following the architectures described below. In this context, encoding may include both source and channel coding.
2. **DNA synthesis** - This phase corresponds to the artificial creation of DNA molecules.
3. **Encapsulation** - This phase corresponds to the storage of the synthesized DNA molecules in a medium to preserve them.
4. **Thermal damage simulation -** This phase targets simulating the degradation that may happen in a real DNA storage system.
5. **DNA release** - This phase corresponds to the extraction of the stored DNA molecules from the storage medium.
6. **Sequencing** - This phase targets determining the nucleic acid sequence, this means the order of nucleotides in the released DNA.
7. **Decoding** - This phase corresponds to the conversion of the DNA units’ sequence back to a convenient form of visual information representation. Eventually, because of robustness issues, error resiliency tools are needed to limit the errors. In this context decoding may include both source and channel

decoding.

DNA media storage may happen according to three basic architectures, notably:

- **Transcoding-based Architecture** - The data to be stored is already available as bits and the DNA coding process corresponds, in practice, to a numerical base transcoding process, notably from base 2 to base 4. This is the most common type of solution reported in the literature, see Figure 2. This type of solution has the following features:
  - May have better integration in the external data ecosystem, e.g. image ecosystem, as the data to be DNA stored (and also read from the DNA store) may be available in an off-the-shelf coding standard, e.g. JPEG, JPEG 2000;
  - Less compression efficient since the important biological constraints are applied to pre-coded data in another numerical base and not directly to the symbols to code;
  - Blind to the characteristics of the input data to code, e.g. its statistics, as only the binary representation is available at DNA coding time;
  - Does not allow control of the quality/rate as the image has been previously coded and lossless only transcoding is happening here.
- **Constrained Coding-based Architecture -** The data to be stored is available as symbols, e.g. quantized coefficients, and the DNA coding process corresponds to a constrained 4-ary coding process (e.g. entropy-based) where the quaternary code is directly created based on statistical information or other constraints, e.g. a fixed length may be used, but also the biological constraints. In this architecture, the top/bottom left bits in Figure 2 should be substituted by some coding symbols, e.g. the quantized transform coefficients from an existing coding standard such as JPEG or JPEG 2000. This type of solution has the following features:
  - May have a poorer integration in the external data ecosystem, e.g. image ecosystem, as the data pipeline does not explicitly offer a binary representation (it may be created by applying binary entropy coding on the decoded symbols); as such there is no full compatibility with existing image coding standards;
  - May consider DNA storage-related constraints on the 4-ary constrained-coding process creating the DNA code;
  - May be more compression efficient since the critical biological constraints are directly applied to the 4-ary coding process without the binary coding intermediate stage; this may be important due to the high cost of the synthesis and sequencing processes;
  - May allow quality/rate control, e.g. in the transform coefficients quantization process.
- **Sample-based Architecture -** The data to be stored is available as component samples, e.g. YUV or RGB samples, and the DNA coding process involves all the modules of the coding pipeline. In this architecture, the top/bottom left bits in Figure 2 should be substituted by component samples. This type of architecture has the following features:
  - Further reduces the interoperability/compatibility with the image coding ecosystem since all the coding modules and not only the constrained/entropy coding module as in the constrained coding-based architecture may be different from available coding solutions; an example could be a deep learning-based codec;

- This architecture only makes sense assuming that it is possible to develop more efficient image coding solutions for the modules before the entropy coding module; however, if this would be true, these advances could be also used for regular image coding standards, reverting this architecture to the constrained coding-based architecture.

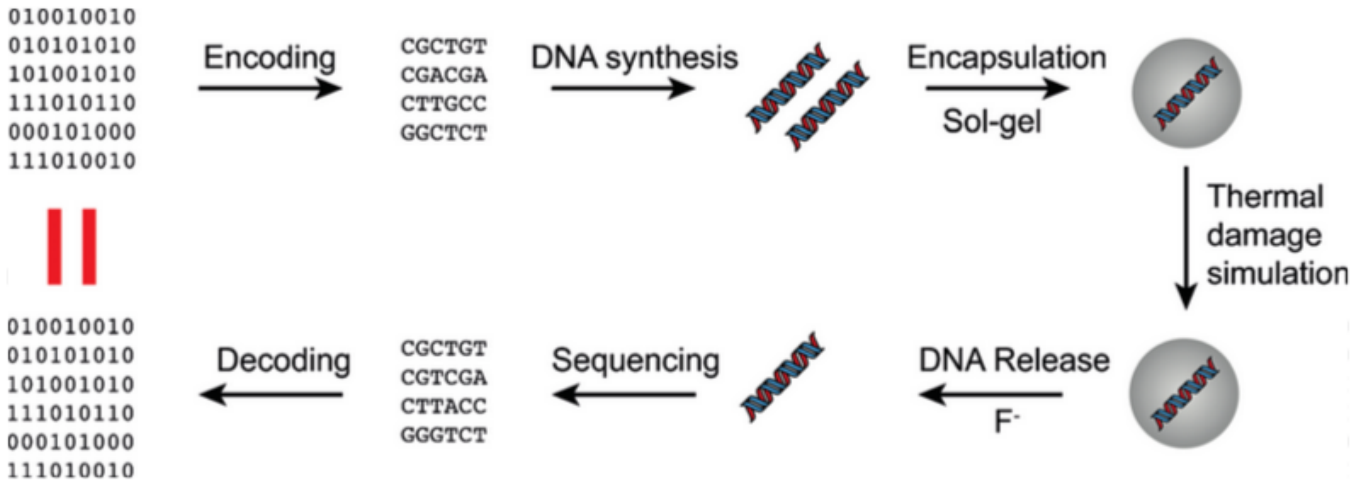


Figure 2 – Transcoding-based DNA media storage architecture [7].

3.2 DNA-based Media Storage: Technology Overview

3.2.1 Introduction

The need for multimedia information storage has increased rapidly over the last few years, calling for further research on novel storage approaches that should allow low-cost, long-term, high-reliability data storage. Among the competing technologies, storage in synthetic DNA strands is very well positioned due to the very high storage density (bits/gram) and long lifetime of the support medium. There are however significant hurdles that need to be overcome to make the technology usable [8], [9] in commercial applications, as described in Section 4 dedicated to the coding-related DNA-based media storage challenges. Figure 3 shows the lifecycle of DNA digital data storage according to [10].

The error rate of DNA synthesis is almost negligible as long as the DNA strands to be synthesised do not exceed the length of 150-300 nts. For longer sequences, the synthesis error increases exponentially. Consequently, to eliminate such errors, the DNA sequences to be synthesized need to be cut into short pieces and formatted in such a way that the initial sequence can be correctly reconstructed during decoding.

In addition, the DNA sequencing biological procedure introduces much error, which cannot be neglected, and therefore there is a need for dealing with the erroneous oligos (short single strands of synthetic DNA) produced by the sequencer. Studies have shown that the three main factors causing errors in the sequenced oligos are:

- **Homopolymers:** Consecutive occurrences of the same nucleotide should be avoided.
- **G, C content:** The percentage of G and C in the oligos should be lower or equal to the one of A and T.

- **Pattern repetitions:** The codewords used to encode the oligos should not be repeated, forming the same pattern throughout the oligo length.

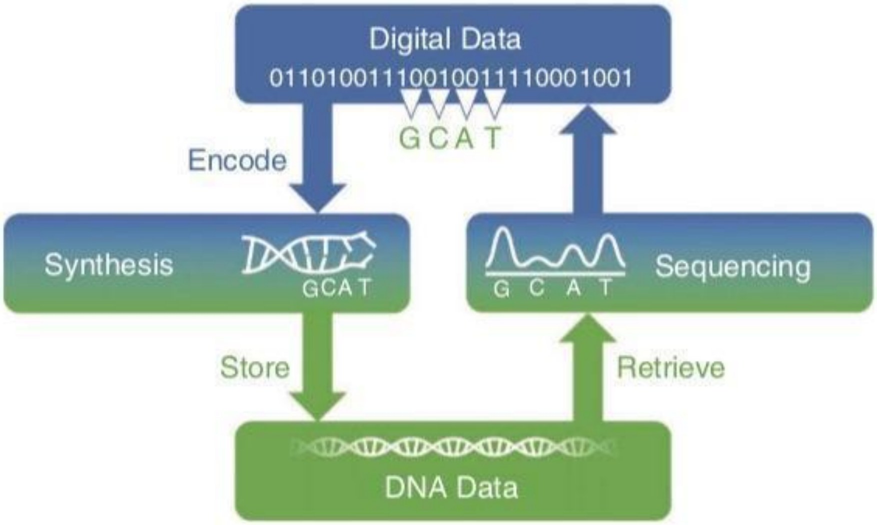


Figure 3 - The DNA data storage lifecycle [10].

Taking into account all the rules above, the sequencing errors can be reduced. Consequently, to be efficient, any DNA coding algorithm should respect the above rules to reduce as much as possible the probability of sequencing errors. This means that the source information has to be adapted to the DNA medium, e.g. by segmenting the encoded source information, applying error control codes and mapping the bits or symbols into the sequence of bases after some coding similar to line-codes used in telecommunications.

Note: Add text to talk about the dependency of some of the above constraints on the approach used for sequencing.

3.2.2 Digital to DNA Mapping

Encoding

It is clear that the encoding of digital data into DNA is strongly constrained by the biological part of the process. More precisely, the encoding should provide a quaternary code that will respect the sequencing restrictions to ensure robustness and the length of the DNA oligos to be synthesized should not be larger than 150-300 nts. Consequently, the structure of a reliable encoder for DNA coding contains the following sub-parts (see Figure 4).

The first step in the encoding workflow is the construction of a dictionary of codewords composed of the symbols A, T, C and G. Those codewords should provide a robust encoding when assembled at a long sequence. This means quaternary strands should not contain homopolymers, high G,C content compared to the content of A and T and, finally, it should not contain repeated patterns. Different codes can be produced each representing the same data based on different optimization criteria and could include efficiency, robustness and cost.

The next sub-process of a DNA workflow is a mapping function, which assigns input symbols to codewords of the quaternary code. This function can be a simple one-to-one function or a more sophisticated one.

Finally, as the oligo length is restricted due to the synthesis limitations to avoid errors, it is necessary to adopt some formatting function for cutting the produced long encoding into shorter oligos and adding special headers for the reconstruction of the input at decoding. Those headers can contain information for the address of the data chunk in the original long sequence, information for any necessary encoding parameters as well as information about the input characteristics as for example its size (see figure 10).

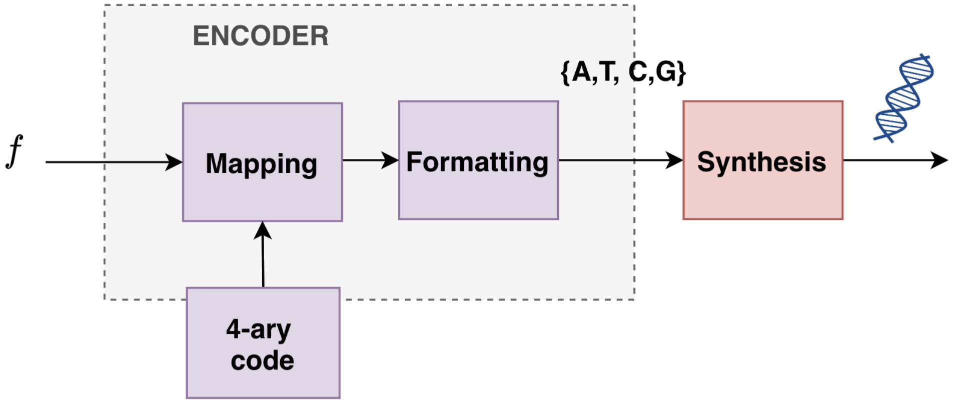


Figure 4 - Encoding of digital data into DNA.

Decoding

Since DNA data storage is a process that is prone to both writing and reading errors, the decoding should include some techniques to predict, detect, or even correct the sequenced data. The addition of redundancy is then necessary for the detection of errors and can be easily achieved using the method of Polymerase Chain Reaction (PCR) amplification, which is applied during both DNA synthesis and sequencing. Consequently, at the output of the sequencer, there will be multiple copies of each synthesized oligo. Each copy might contain different types of errors in various positions and this yields the need for selecting the most representative copy for each oligo. This selection can be based on computing a consensus sequence using all of the erroneous copies of each oligo or on finding the most frequent among all copies. This process can be followed by some error correction algorithm to treat any remaining errors to obtain an error-free decoding. It is important to mention that the efficiency of the error correction highly depends on the techniques and machines that have been used during sequencing as some particular sequencers can cause higher error rates than others and can therefore create stronger distortion. Finally, using the inverse mapping function one can retrieve the digital information which had been stored into DNA. An overview of the decoding process is presented in Figure 5.

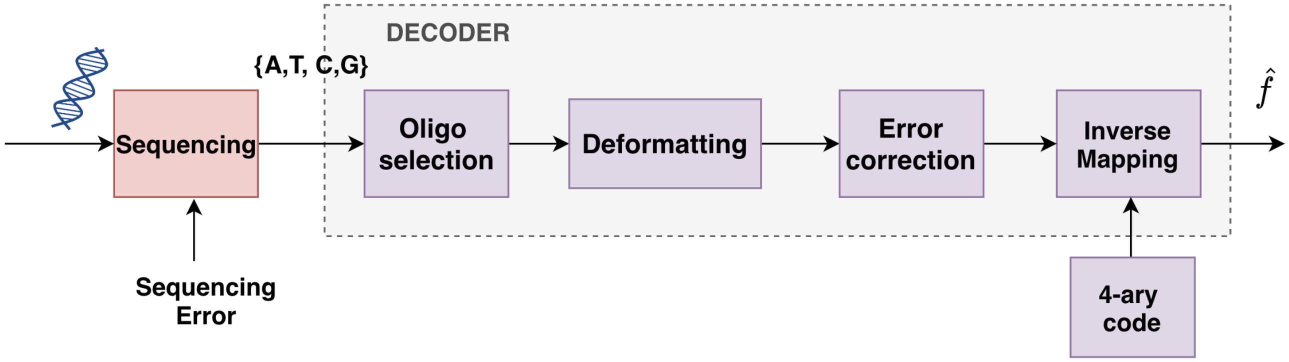


Figure 5 - Decoding of digital data stored into DNA.

3.2.3 Existing works

DNA data storage is a relatively new field of research and thus the state-of-the-art is limited to a few pioneering works which have however contributed widely to this emerging topic.

The first application of DNA data storage

In 2012, George Church *et al.* encoded for the first time a 659-Kbyte book that was co-authored by Church into DNA. In their experiment, the authors used a very simple encoding, by randomly translating zeros to A or C and ones to T or G [8]. The encoded sequence was then written onto a microchip as a series of DNA fragments using an ink-jet DNA printer. The encoding resulted in 54,898 oligonucleotides, containing 96 bases of data along with a special 22-base sequence at each end to allow the fragments to be copied in parallel using the PCR amplification, and a unique, 19-base “address” sequence to denote the segment’s position in the original document.

The resulting PCR amplified oligos were then read back using an Illumina sequencer to retrieve the original text. The storage density of the DNA fragments produced by this method was estimated to be more than 700 terabytes per cubic millimeter. This result represented the largest volume of data ever artificially encoded in DNA and proved that data density for DNA is several orders of magnitude greater than that of state-of-the-art storage media.

This work made a pioneering step to prove the feasibility of using DNA as an alternative means of storage while also demonstrating the extraordinary capacity compared to conventional storage devices and revealing that sequencing may be an error-prone process. By analyzing the different errors which occurred during sequencing, this work provided a first study of the main constraints to be respected during the encoding.

First biologically constrained DNA encoding

In 2013, Goldman *et al.* [6] proposed a novel algorithm for encoding digital data into binary while respecting the main sequencing constraints. The encoding was proposed using a ternary Huffman algorithm to encode each byte of a binary sequence into the digits 0, 1, and 2. Those digits are then associated with three of the symbols A, T, C, and G, omitting the symbol that has been used for the encoding of the previous digit, so as to ensure that no base is used twice in a row. This strategy avoided the creation of homopolymers while still making use of DNA’s four-base potential. To enhance the reliability of the oligos and determine the data's position in the original file, Goldman’s team synthesized oligonucleotides carrying 100 bases of data, with an overlap of 75 bases between adjacent fragments, so that each base was represented in four oligonucleotides creating a four-fold redundancy. Even so, the researchers lost two 25-base stretches during sequencing, which had to be manually corrected before decoding. The encoding followed in this study is explained in Figure 6.

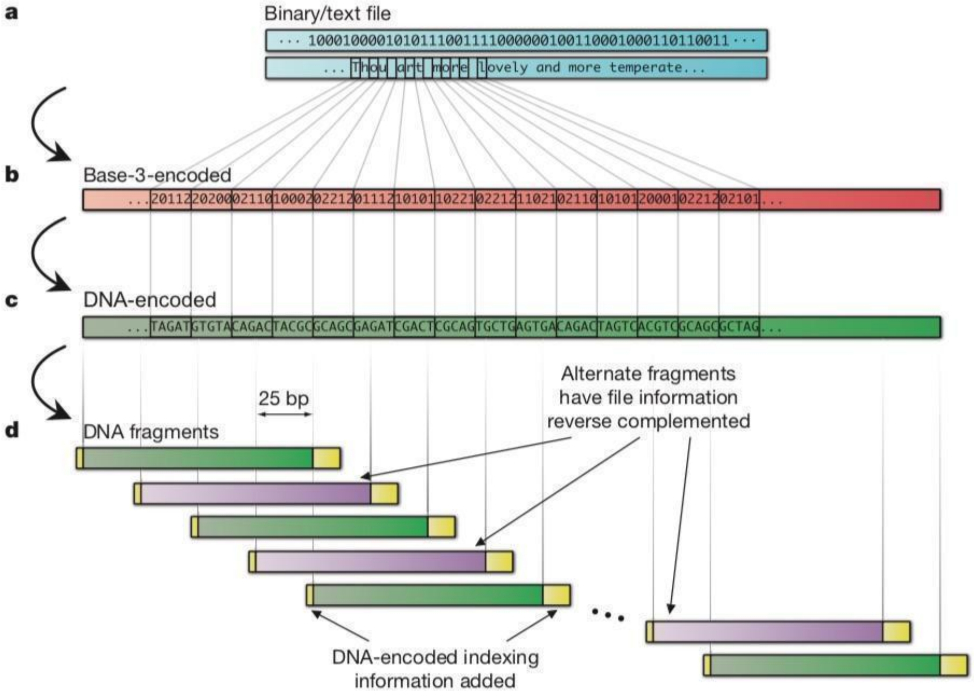


Figure 6 - Digital encoding in DNA [10].

Introduction of Reed-Solomon codes for DNA encoding

To deal with the remaining sequencing errors, in 2015, Grass and his team [11] have proposed for the first time the use of Reed Solomon codes to introduce error correction in the encoding. More precisely, in this work, the authors proposed mapping the data to blocks containing elements from Galois Field 47 (GF(47)). The column of each block is extended using a unique index consisting of elements in GF(47). The extended columns are then encoded to DNA by mapping each of the GF(47) elements to a triplet of nucleotides while ensuring that there is no repetition of the same base in the two last positions, thus guaranteeing that homopolymers are avoided. Each encoded column represents a DNA fragment to be synthesized and stored in silica to ensure long-term storage without DNA corruption. In their study, the authors reported perfect retrieval of 83 kB of data encoded using a Reed-Solomon code, an error-correcting code used in CDs, DVDs, and some television broadcasting technologies. The storage workflow is shown in Figure 7.

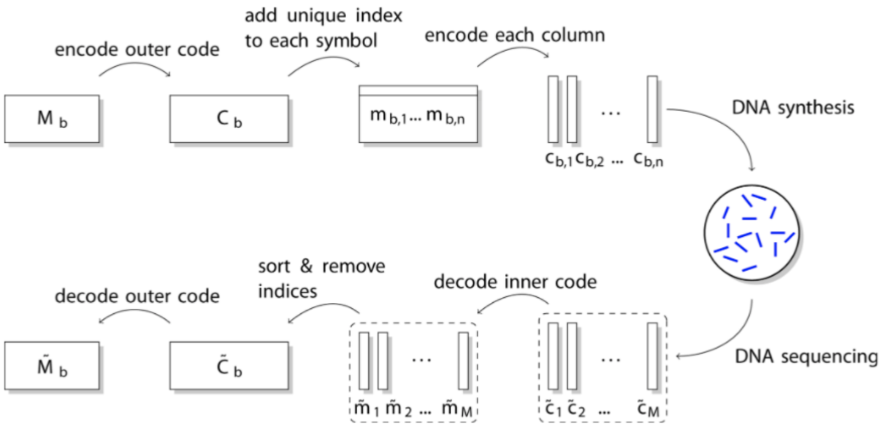
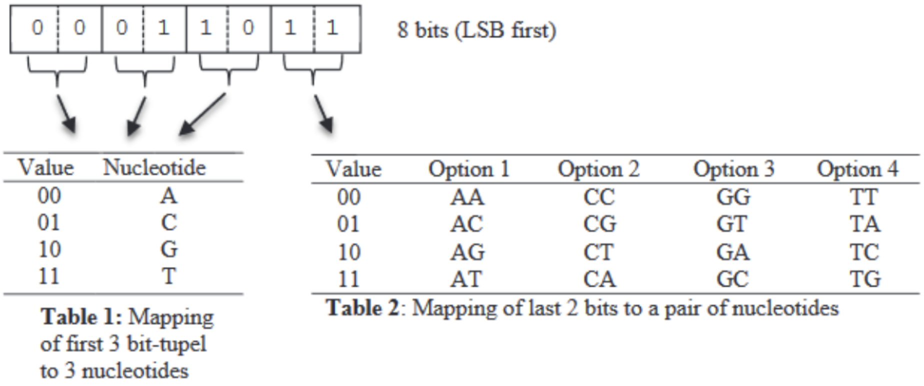


Figure 7 - Grass et al. encoding [11].

In 2016, Blawat *et al.* [12] proposed another interesting method for constructing a robust quaternary code by encoding each byte of some digital data to 5 nucleotides using the following algorithm. To begin with, the first three pairs of bits are encoded to the corresponding nucleotides from table 1 in Figure 8 and represent the first, second, and fourth nucleotide, respectively, in the encoded DNA word. Then the last pair of bits can be encoded to a pair of nucleotides among four different options as presented in Table 2 of Figure 8 and will be placed in the third and fifth position of the resulting DNA word.

As a result, for each byte, four different DNA words are provided. To ensure that the limitation concerning the maximum run length is respected, the four options are filtered so as not to create homopolymers.


0 0 0 1 1 0 1 1
8 bits (LSB first)

Value	Nucleotide
00	A
01	C
10	G
11	T

Table 1: Mapping of first 3 bit-tupel to 3 nucleotides

Value	Option 1	Option 2	Option 3	Option 4
00	AA	CC	GG	TT
01	AC	CG	GT	TA
10	AG	CT	GA	TC
11	AT	CA	GC	TG

Table 2: Mapping of last 2 bits to a pair of nucleotides

Figure 8 - Blawat et al. encoding [12].

To do so, the authors propose keeping only the options that do not violate the following rules: i) the first three nucleotides shall not be the same; and ii) the two last nucleotides shall not be the same.

With the above-described constraints, at least two valid DNA symbols can be found for every data byte, thus introducing some redundancy, which can be used for error detection. More precisely, the authors proposed separating the different codeword options into different predefined clusters and encoding each input byte using the encoding of a specific cluster, according to the byte's position. For example, one option would be to use codewords from cluster A to represent even byte positions and from cluster B to represent odd byte positions. Thus, in the case where an error alternates a codeword expected to be found in one cluster to another one that belongs to some other cluster, error detection is possible. Furthermore, in this work the authors propose robustifying the addressing headers using Reed Solomon codes to allow a more reliable decoding.

First robust random-access implementation

In the same year (2015) Yazdi et al. [13] introduced an important way for allowing random access using specific and robust addressing in the encoding. In their study, the authors proposed the addition of some specially designed primers in both ends of the encoded data to allow selective PCR amplification of particular oligos instead of amplifying the full oligo pool. The primers were specially designed to be robust to sequencing errors and the encoding DNA words for each oligo depend on the corresponding primer. More precisely, for each oligo the DNA code is constructed by ensuring there is no correlation of the payload to the oligo's addressing primer as this would create secondary structures, which can be catastrophic and can lead to losing the full oligo during sequencing.

In a later study published in 2017 the authors provided an experiment testing the efficiency of their proposed encoding using the MinION—Oxford Nanopore's handheld sequencer for the reading of the DNA while also using JPEG compression to reduce the synthesis cost. This study has devised error-correcting algorithms specifically for the kinds of mistakes the MinION makes.

DNA coding using Fountain codes

Still in 2016, Columbia University researchers Yaniv Erlich and Dina Zielenski proposed a method based on a Fountain code [14], an error-correcting code used in video streaming. As part of their method, they use the code to generate many possible oligos on the computer, and then screen them in vitro for desired properties. Focusing only on sequences free of homopolymers and high G content, the researchers encoded and read out, error-free, more than 2 MB of compressed data—stored in 72,000 oligonucleotides—including a computer operating system, a movie, and an Amazon gift card. This encoding architecture is presented in Figure 9.

First, the input binary file is segmented in partitions. Then, using a luby transform, droplets of bits are created by selecting randomly segments from the input sequence and bit-wise adding them, attaching also the random seed used for the selection. The resulting bit droplets are then encoded into quaternary symbols and scanned for satisfying the biological constraints of GC content and homopolymers. Encoded droplets which do not respect the above restrictions are discarded while the rest are used for creating the oligos. This process is repeated until enough oligos are produced resulting in a densely compressed encoding, reaching a capacity of 1.98 bits/nt.

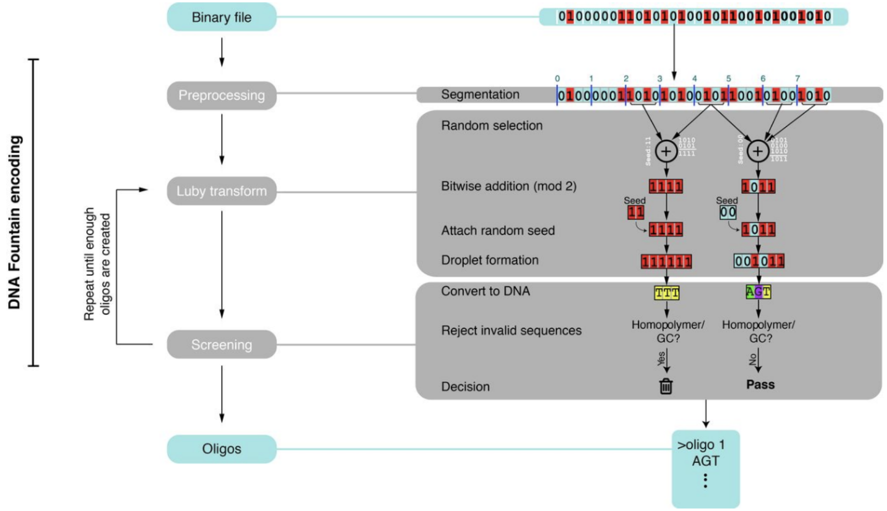


Figure 9 - Erlich et al. encoding [14].

Efficient end-to-end DNA coding workflow

In 2016, Borhholt *et al.* from Microsoft Research proposed a DNA based archiving system using the quaternary code introduced by Goldman *et al*. In this study, they improved the encoding by avoiding the fourfold redundancy using themselves addressing primers for allowing random access [15]. Later in 2017, researchers at Microsoft Research presented some extra studies to improve their results using a clustering algorithm to cluster and correct the multiple reads provided by the sequencer, allowing a better reconstruction quality [16], [17]. Finally, in 2019, a Microsoft Research team successfully encoded the word “hello” in snippets of fabricated DNA and converted it back to digital data using a fully automated end-to-end system described in [18].

Closed loop optimization of DNA image coding

All the above studies provide some way for building a quaternary encoding of digital data by respecting the biological constraints. Each one of those encodings exhibits different advantages and weaknesses and, since the subject is still very new, it is necessary to provide new encoding ideas, which can help enrich existing solutions and improve the quality of the stored data.

As the main DNA data storage drawback is the high synthesis cost, the encoding methods proposed in the literature attempt to improve the storage capacity while also being robust to sequencing errors. To this end, some studies have proposed compressing images with JPEG before encoding. However, no study has proposed a method for controlling this compression such that it provides a closed loop solution, which can allow selecting the best compression parameters for a given coding potential. In the recent study of Dimopoulou et al [19], [20], a source allocation algorithm was developed which offers the possibility of not only reducing the synthesis cost, but also promising an optimal quality of the stored image for a predefined encoding rate and thus a given synthesis cost.

As a low complexity source allocation requires a fixed length code, they also proposed a new efficient algorithm for the construction of a robust fixed length DNA code that facilitates the nucleotide allocation method. Then, two different mapping methods were introduced. The first deals with pattern repetitions which might be the cause of error increase in the Illumina sequencers and has not been tackled by previous studies; the second aims at decreasing the visual impact of substitution errors which may remain after error correction] [18], [21], [22].

Finally, a new formatting structure was presented for cutting the encoded information into oligos and adding the needed headers, which suits the proposed encoding.

3.2.4 Segmentation and Reassembly

Synthesizing long chains of DNA was/is challenging and long chains are prone to single and multiple base errors and erasures. To address these problems, most works on DNA data storage rely on short DNA chains to represent the data, mandating the use of segmentation of the data prior to mapping/synthesis into DNA strands. The original order of the data can be recovered if some kind of addressing or indexing is used to signal the segments order. More complex and higher-level indexing schemes can be used as shown in Figure 10 which depicts a DNA fragment format used in an image DNA storage method [19] that includes primers end markers as well as an ID field used to identify images.

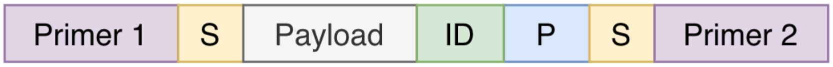


Figure 10 - Dimopoulou et al DNA packet format [19].

3.2.5 Characterization of the DNA Data Storage Channel

Although applying robust and efficient encoding techniques is highly important for DNA data storage so as to reduce its cost and maximize its reliability, the process involves some biological procedures which can corrupt the DNA molecules. Synthesis, sequencing, storage and the manipulation of DNA (mainly PCR amplification) may introduce imperfections in the DNA strands and cannot be ignored when designing DNA storage systems as they jeopardize the integrity of the stored content.

3.2.5.1 Source of Errors

Synthesis

The error rate of DNA synthesis is almost negligible when the synthesized oligos do not exceed 300 nts and increases exponentially for bigger lengths. The noise introduced can be in the form of insertions, deletions, substitutions and early termination, which is an extreme case of deletions. Early termination depends on the synthesis method and the position within the DNA sequence and occurs when the nucleotides can no longer be added to the growing strand due to the loss of chemical reactivity at its end. Additionally, not all the growing strands corresponding to the same target sequence face the same errors, meaning that there might be different variations for every reference. Finally, depending on the selected synthesis technology, the number of copies generated per reference may vary, leading to an uneven distribution of the synthesized strands [35].

Polymerase Chain Reaction

PCR amplification is commonly used to increase the redundancy in the DNA strands. However, the addition of redundancy is not its sole purpose. As the process of PCR requires the presence of primers in both ends of the amplified sequences, when performed in non-complete sequences (e.g. broken strands or sequences that suffered early termination during synthesis) those strands will not be amplified and thus, cleaned from the oligo pool. Despite being a reliable process, PCR introduces a certain bias in the distribution of the number of copies as this process has a predisposition towards certain sequences over others. As an example, the number of generated copies is smaller for sequences with a high GC content [35].

Storage

DNA is prone to chemical decay if not stored under optimal conditions. Its degradation can lead to the loss of entire DNA molecules, which ultimately alters the distribution of the molecules in the pool. DNA decay has a minimal effect on the probabilities of insertions and deletions but significantly increases the substitution rate. The estimated half-life of fossilized DNA is around 500 years but it can be increased if stored in a controlled environment. There are several methods for the storage of DNA and the selection of the optimal one will depend on the frequency of access. [35]

- Long-term storage (accessed every +10 years)

  One of the main applications of long-term storage is the preservation of 'cold' data as it is the case of historical records, legal evidence or cultural patrimony. In this context, DNA can be stored in its dehydrated form and embedded into silk matrices, salts or even encapsulated. Some studies have shown that the encapsulation of DNA preventing its contact with water and oxygen provides the stored DNA a half-life at room temperature of approximately 170,000 years [33]. However, determining the stability of DNA in the long term remains a challenge as the methods to measure the degradation are not sensitive enough in most of the cases. Additionally, aging models are usually dependent on temperature but it remains unclear the existence of other degradation mechanisms which do not have dependency on temperature. [36]

- Medium-term storage (accessed multiple times per year)

  Although DNA encapsulation is a promising solution for the storage of DNA which allows to maintain its stability for hundreds or even thousands of years, it is not efficient when it comes to data that has to be accessed every few months as its physical manipulation is delicate and time-consuming. Instead, semi-accessible forms of storage are used for such cases as for instance refrigerated or frozen in aqueous solution and dry solid. Studies show that the stability of DNA stored at 4°C or below (either in aqueous form or dried) can last for around 2 years. However, the main challenge lies in the amount of degradation that occurs every time the information has to be accessed. Every time the solution containing the DNA is frozen or thawed, ice crystals are formed increasing the probability of strand breakage, which increases with the length of the strands. [36]

- Short-term storage (dynamic handling)

  In-storage computation merges DNA-based computation and DNA storage systems. It has the potential of allowing direct search and edit of DNA and promises to lower the latency of conventional systems. However, such emerging systems will require the physical manipulation of the DNA molecules stored in a soluble aqueous form which risks the integrity of the DNA strands (e.g. strand breakage). Furthermore, although the biochemical environment of the molecules can be controlled to keep the optimal conditions to increase DNA stability, some biological manipulations involved in in-storage computation, editing or PCR amplification require temporary exposure to high temperatures increasing the probability of DNA degradation. [36]

Sequencing

Since the release of nanopore sequencers, this technology has become more and more popular thanks to its affordability, small size and speed, which make it suitable for real-time applications. More precisely, nanopore-based sequencers measure the changes in the electrical conductivity as DNA strands pass through the pore. This electrical signal is then translated into a sequence of nucleotides in a process known as basecalling. Despite all the advantages that sequencers such as the MinION offer, it has a major drawback as it remains an error-prone process. This constitutes the main challenge when using this device in the context of DNA data storage, compromising the decodability of the data.

The noise is introduced in the form of insertions, deletions and substitutions during the basecalling process and the error rate increases if the sequence contains homopolymers (>5 nts), pattern repetition or/and high GC content. According to some studies, the noise introduced by nanopore sequencers dramatically affects both ends of the DNA strands but it remains unclear how the length of the input sequences affects its performance.

3.2.5.2 Simulators

In the past years, several works have introduced sequencing simulators aiming to ease the implementation of new algorithms targeting nanopore-sequenced data. Such simulators allow testing while developing new tools thanks to their speed, low cost and high throughput. Commonly, simulators generate noisy reads using a model error profile extracted from experimental data. The introduction of the errors can be done directly by modifying the bases of the DNA sequences or by simulating the electrical signals and allowing the basecaller to introduce the errors while translating it into a sequence of nucleotides which provides a more realistic scenario. The main challenge when using these simulators for DNA data storage applications lies in the fact that their error models are generated from the sequencing of complete genomes and thus, longer reads than in the case of synthetic DNA, whose length is limited to 300 nts.

Some of the latest open source simulators are described in the following paragraphs:

- NanoSim (2017) [37]

  This nanopore sequencing simulator models errors from 6 datasets using different generations of MinION sequencing kit as statistical mixture models. It samples the input reference and generates reads with specific length distribution. Each of those reads are then aligned to the training genomes. In case an alignment is found, the simulator will add errors from the mixture model corresponding to the genome, otherwise, if no alignment is found for the reads, it will add an arbitrary high error rate compared to the aligned reads.

  Source code: https://github.com/karel-brinda/NanoSim-H.git

- DeepSimulator (2018) [38,39]

  DeepSimulator is a deep learning based simulator able to generate reads with almost the same properties as the real data. The workflow could be divided into 3 main modules:

1. Sequence generation module: Given the user-specified reference genome, as well as the number of reads to be generated, the sequence generation module randomly chooses a starting position in the reference sequence to produce the relatively short sequences, which satisfy the length distribution of the experimental Nanopore reads.
2. Signal simulation module: the pore model takes as an input a nucleotide sequence and outputs the expected current signal for each 6-mer (subsequence of 6 nucleotides) in the sequence. Then, random Gaussian noise is added according to the user-defined variance parameter to produce the simulated signals.
3. Basecaller: same as in the Nanopore sequencer.

What makes DeepSimulator closer to the real sequencer is the fact that it does not explicitly introduce substitutions, insertions or deletions directly at the read level as is usually done in the rest of available simulators. Instead, it mimics the electrical signal produced by Nanopore sequencing as similar as possible and then, it is the basecaller that introduces the errors by itself as it happens in the real sequencing procedure.

Source code: https://github.com/liyu95/DeepSimulator.git

- MESA (2020) [40]

To our knowledge, MESA is the only currently available simulator able to model the full DNA storage channel, including synthesis, PCR amplification, storage and sequencing. It uses either published error profiles or user-defined rates. The sequencing module allows to simulate reads from different sequencing platforms such as Illumina, PacBio and Nanopore. In addition, this tool offers the possibility of assessing the quality of the DNA fragments regarding their content (i.e. whether the biological constraints are respected or not)

Source code: https://github.com/umr-ds/mesa_dna_sim

Web application: https://mesa.mosla.de/

Evaluation metrics: Commonly, the reliability and performance of sequencing simulators is measured by comparing the introduced error rates (substitutions, insertion and deletions) and length distribution of the output reads to the empirical ones. In the case of simulators that mimic the electrical signal measured by the nanopore sequencer, the average Dynamic-Time-Wrapping (DTW) deviation between the simulated signals and the empirical ones is often used to measure the reliability of the simulated signals. Additionally, some comparative studies evaluate the simulators based on precision and recall [41]. They are computed by mapping the output reads to the reference and checking either the mapping is correct or not. In this context, precision measures the fraction of correctly mapped bases out of the total number of mapped bases and recall is defined as the fraction of correctly mapped bases out of the total number of bases in the reads.

3.2.5.3 Error Control

To retrieve information stored in DNA, first PCR (Polymerase Chain Reaction) has to be employed to multiply the DNA strands to reach numbers beyond the detectability thresholds of the equipment in charge of the next step, i.e. sequencing. After PCR, multiple copies of each strand, possibly with errors, are aligned and, as illustrated in Figure 11, some sort of voting or parity scheme is used to obtain the error-corrected strand.

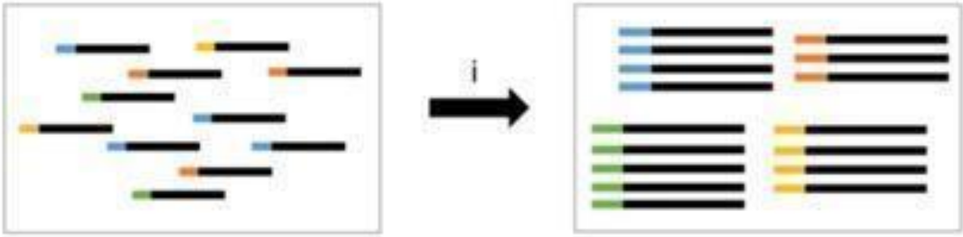


Figure 11 - Strand alignment during sequencing [23].

More sophisticated methods for error control can be used, such as Reed-Solomon codes, applied as suggested by [23] and shown in Figure 12.

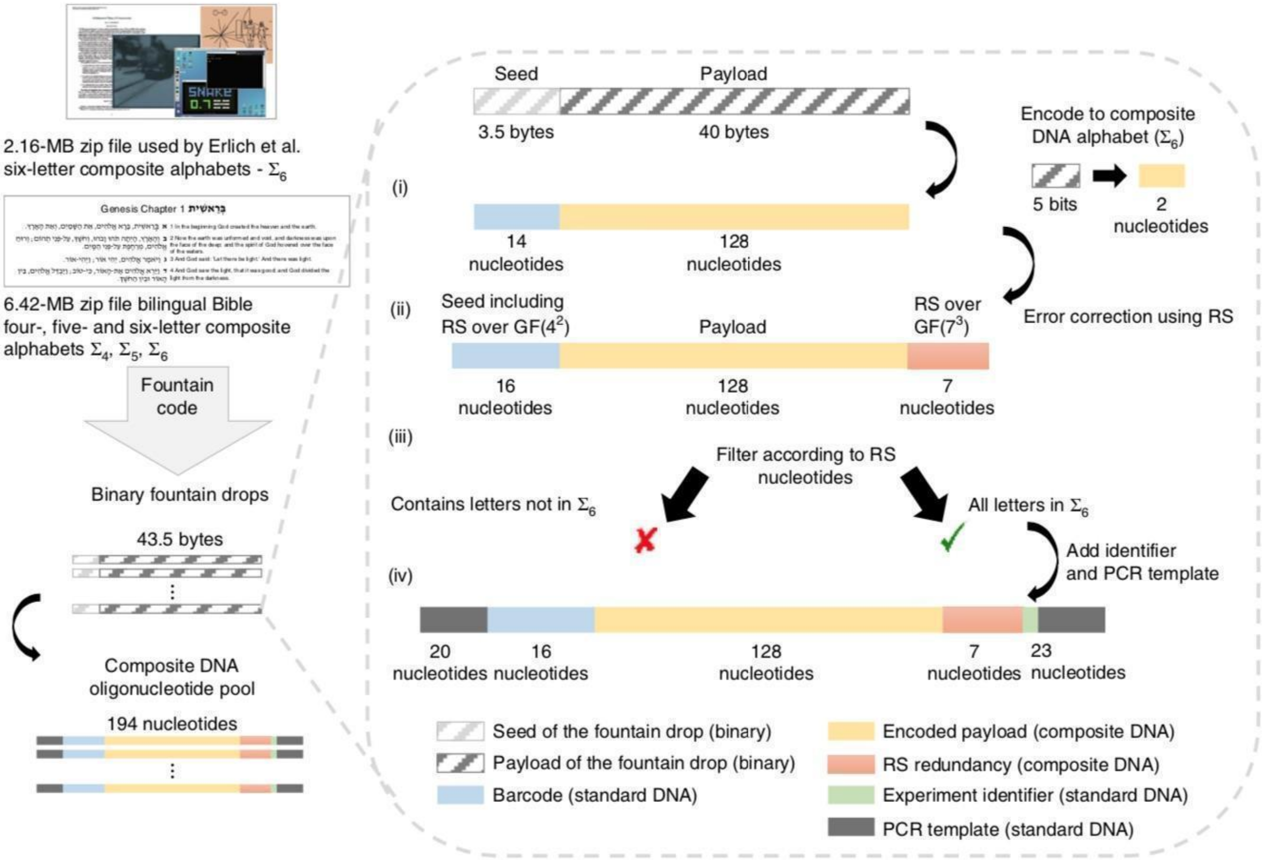

2.16-MB zip file used by Erlich et al. six-letter composite alphabets - Σ_6
6.42-MB zip file bilingual Bible four-, five- and six-letter composite alphabets Σ_4, Σ_5, Σ_6
Fountain code
Binary fountain drops
43.5 bytes
Composite DNA oligonucleotide pool
194 nucleotides
Seed
Payload
3.5 bytes
40 bytes
(i)
14 nucleotides
128 nucleotides
Encode to composite DNA alphabet (Σ_6)
5 bits
2 nucleotides
(ii)
Seed including RS over GF(4^2)
Payload
RS over GF(7^3)
Error correction using RS
16 nucleotides
128 nucleotides
7 nucleotides
(iii)
Filter according to RS nucleotides
Contains letters not in Σ_6
All letters in Σ_6
Add identifier and PCR template
(iv)
20 nucleotides
16 nucleotides
128 nucleotides
7 nucleotides
23 nucleotides
Seed of the fountain drop (binary)
Payload of the fountain drop (binary)
Barcode (standard DNA)
Encoded payload (composite DNA)
RS redundancy (composite DNA)
Experiment identifier (standard DNA)
PCR template (standard DNA)

Figure 12 - Encoding pipeline using error-correcting coding [23].

3.2.6 Multimedia Storage in DNA

An early effort to store audio information (music) in DNA was made by Twist Bioscience, Microsoft, the University of Washington, EPFL, and the Montreux Jazz Digital Project as reported in [24]. This project recorded both Deep Purple’s “Smoke on the Water” and Miles Davis’ “Tutu” songs in DNA, “making scientific history.”

At least two types of approaches can be used for image data storage in DNA. The simplest one involves storing bitstreams representing images obtained by the use of e.g. JPEG encoders, using the DNA storage procedures summarized before. However, to ensure better adaptation to the characteristics of the storage medium, i.e. DNA, and possibly achieve higher storage efficiencies, it is better to design coding algorithms specific for DNA storage. Risking leaving out other relevant works, the methods proposed by Dimopoulou et al. in 2018, 2019 and 2020 [5], [19] [20] should be cited. The solution described in [19] is particularly interesting as it is based on a DWT image decomposition where the DWT coefficients are scalar quantized and an optimal nucleotide allocation is employed to minimize the distortion values and to constrain the length of the nucleotide strand for each sub-band given by the encoder. This allocation affects the choice of the quantization step size. The nucleotides generated are then transformed to synthetic DNA, after splitting into smaller segments, usually with less than 150 nucleotides, to control the sequencing error rate. The fragment reassembly is made possible by the addition of headers to the oligos as shown in Figure 7. The headers contain the localization of each split segment encoded information, allowing further information recovery and decoding. Moreover, the stored data is also amplified, thus creating several copies using PCR to deal later on with sequencing errors. An early and simple example of applications of DNA storage to encode movies is briefly described in [25].

4. Coding-related DNA-based Media Storage Challenges

The main coding-related challenges in DNA-based media storage are:

1. **DNA-based writing/synthesis and reading/sequencing costs** - While the cost of DNA-based writing and reading are currently prohibitive for large amounts of data, it has been reducing and it may be affordable in the future, at least for specific applications scenarios. This cost is not only dependent on the amount of data, as different sequences tend to have different production costs. Some symbol combinations might be relatively simple to synthesize, while other combinations might result in prohibitive prices. Hence, the cost depends on the content and the coding solution.

2. **DNA-based writing/synthesis and reading/sequencing speed** - The DNA-based writing and reading processes are currently slow. However, as it happens with any storage technology, this current limitation might tend to vanish in time.

3. **Biological-related constraints** - The coding processes defining the sequence of DNA bases to be used have to consider the relevant limitations and constraints in terms of DNA bases combinations while maximizing the stored data density. Two types of constraints can be defined:

3.1 **Sequencing constraints -** Some sequences are impossible to produce (like four repetitions of the A bases) or might grow the production costs. Moreover, some sequences are prohibited because they correspond to non-artificial content, leading to the production of existing living cells, virus or even non existing living cells.

3.2 **Error-related constraints -** The biological properties of the nucleic acid and the molecular machinery used to read and write may create errors specific of this technology; for example sequences containing lots of G nucleotides are difficult to write, for example, because they often produce secondary structures that interfere with synthesis [26]. .

4. **Random access** - The basic storage processes are not amenable to random access, which requires special

attention as it is a fundamental coding related functionality; in this case, it is critical to be able to read a part of the data without having to read the full data. Currently, the data needs to be prepared so the DNA molecules are generated at the storing time. After that, it is not possible to add new data, nor is it possible to remove data from the initially stored. These types of constraints were also present when technologies like ROM or CD/DVD were developed, but they were alleviated with time.

5. **Biosafety and ethical issues** - Because dealing with the code of life, there are challenges related to the creation of dangerous DNA sequences, notably corresponding to known or unknown viruses or other dangerous pathogens. All the safety and ethical issues related to this possibility need to be carefully addressed, clarified and avoided.

In summary, considering the current technological status quo, appropriate trade-offs between cost, complexity, compression efficiency, and error resilience needs to be found to allow the future deployment of DNA based media-storage.

5. DNA-based Media Storage: Use Cases

DNA-based representations of media data might provide efficient means for storage of huge data. Synthetic DNA provides a very high storage density compared with the traditional electronic and magnetic based methods. Furthermore, it also provides long-term support for data, which is not comparable with the traditional storage devices. According to [19], DNA has the theoretical ability to store more than 450 Exabytes in 1 gram, which is not comparable with current HDD technology that requires 600 grams for a 10TB storage. Moreover, DNA can last for centuries, which is not comparable with the typical duration of the current storage devices. Finally, it is becoming fast, easy and cheaper to perform in-vitro replications of DNA. In fact, DNA-based storage is considered as one of the solutions to the growth of digital data that some believe to reach over 170 zettabytes in 2025 [10]. Most of this data is related to the proliferation of media information over social networks. However, most of this media information is almost never accessed (the so-called cold data) and its storage does not require very efficient access. Currently, DNA still faces the lack of random access which limits efficient access times.

While storage is the key denominator, there are different relevant use cases depending on specific requirements in terms of storage longevity, target quality, etc.

5.1 Long Term Media Archives and Cultural Heritage Preservation

Considering the complexity of the storing/synthesis and reading/sequencing processes, DNA-based storage seems to firstly target large scale, long-term preservation archives with DNA-based storage confined to one or a few central storage units where information is only intended to be accessed infrequently [26]. In this case, longevity is a key requirement and no quality degradation seems to be acceptable. Lossless coding may also be a relevant requirement. National archives of audiovisual media and cultural heritage artefacts clearly fit in this use case. Cultural heritage artefacts archival covers a wide field of types of digital items from scanned ancient books, maps and photos to three-dimensional models of small objects like statues, flat objects like textile samples and entire ancient carpets, zoology and geologic specimens and ancient scientific apparatuses. According to archive curators besides high fidelity coding, scalability and random access should be supported for efficient browsing and detailed inspection. Storage of metadata jointly with the artefacts records is also seen as very important, as it is common practice to have artefact expert analysis and commentary stored as metadata.

5.2 Social Networks Cold Media Storage

With the explosion of social networks, huge amounts of personal media data are created, which should be stored for long periods, e.g. the lifetime of users. However, most of this data, getting old with time, are infrequently accessed, thus justifying the so-called cold storage. In this case, some quality degradation may be acceptable over time. Companies like Facebook, Instagram, etc, may fit in this use case.

5.3 Preservation of Medical Images

Medical images are typically represented by huge amounts of data. Several medical diagnosis systems generate images that require very high resolution and high dynamic ranges, while multiple systems represent volumes through several longitudinal slices. In multiple medical image applications, lossy storage is not acceptable. One of the reasons is related to legal issues that can arise if a wrong diagnosis exists, that could be considered as caused by the compression technology. Furthermore, radiologists and physicians of some specific specialities are trained to use almost imperceptible textures for diagnosis, that should not be affected by any lossy coding mechanism.

Nowadays, because of the huge storage requirements, hospitals usually just delete the medical records of patients after some time, typically after the patient's disease. This behaviour limits the possibility of this information being used in follow up studies or as case studies.

DNA-based storage would be a natural option for medical images that could solve this storage problem. Furthermore, hospitals typically have special conditions for DNA preservation, which makes the DNA storage a perfect solution for medical imaging long term storage. Due to the type of data in question, lossless coding and high error resilience are thus critical.

5.4 Preservation of Large-scale Repositories of Biomedical Data: Beyond Local Data Storage

The availability of a standardized effective compression and coding strategy would be essential for setting the basis for a unified framework for the collection, sharing and processing of big biomedical data. Medical data are mostly acquired and represented in higher dimensional spaces: from volumetric data, such as structural MRI and CT, to 3D+time, such as functional MRI, where the acquisition consists of a temporal series of 3D volumes, to 4D, as it is the case for diffusion weighted MRI (dMRI) where many volumes are acquired in a single scan, ranging from 32 for basic clinical acquisitions to 256 or more for advanced acquisition schemes. The exploitation of the native dimensionality of the data would lead to remarkable improvements in compression performance also in the lossless regime. This would follow the spirit of JP3D, the extension Part 10 of JPEG 2000 targeting volumetric data, expanding to higher number of dimensions and accounting for inter-volume-redundancy and potentially leading to a tremendous gain in compression. In addition, ensembles of multimodal data are increasingly available for the same subject. The exploitation of intra-subject correlations across modalities could in principle be also exploited.

Metadata allowing to effectively retrieve the whole set of information concerning a specific query, for instance, a given subject or a given feature (type of pathology, acquisition site, etc) would be of great help for the research community. Indeed, such big data repositories are multimodal (including imaging data from many different acquisition modalities), multiscale (including genetic, imaging, behavioral and lifestyle data), and multidimensional, ranging from 1D (neurophysiological signals, genetic data), to 2D (images), 3D (structural MRI, CT etc), and 4D (volume sequences such as diffusion MRI, functional MRI, ASL), and an effective indexing policy allowing random access and fast retrieval is still missing. Needless to say, data from gene sequencing and expression is growing exponentially, and the DNA coding strategy could be a very elegant solution to the long-term storing of such data.

The increasing need of large-scale studies have boosted a number of initiatives targeting multicentric data collection and sharing calls for new and efficient means of long-term data storage, even before pandemics. Standardization comes naturally into play in order to facilitate and enhance data sharing and algorithm benchmarking.

Remarkable examples are the UK BioBank (https://www.ukbiobank.ac.uk/), collecting a large-scale biomedical database and research resource, containing in-depth genetic and health information from half a million UK participants, the Alzheimer's Disease Neuroimaging Initiative (http://adni.loni.usc.edu/), including MRI and PET images, genetics, cognitive tests, CSF and blood biomarkers as predictors of the disease of both healthy and diseased subjects, the Human Connectome Project (http://www.humanconnectomeproject.org/), that represents the first large-scale attempt to collect and share data of a scope and detail sufficient to begin the process of addressing deeply fundamental questions about human connectional anatomy and variation of healthy individuals. In addition, standardized effective coding would support initiatives such as ENIGMA (Enhancing Neuro Imaging Genetics through Data Analysis), a consortium bringing together researchers in imaging genomics to understand brain structure, function, and disease, based on brain imaging and genetic data.

5.5. DNA Coding for Traceability

One important use case where DNA data storage could be used is for traceability purposes. In other words, the ability to identify and trace elements of a product as it moves along the production chain from raw material to finished products. Traceability is highly important as it can allow easy identification of produced products revealing information which is fundamental for protecting consumers and increasing their trust on different brands. Traceability is the process of marking products with some special barcode containing all information regarding the product's authenticity by listing all the ingredients as well as the ingredients' origin and characteristics. This barcode should therefore follow the product throughout its lifespan.

When traceability is applied on materials such as textiles, gold, diamonds, or construction materials such as concrete, it is essential that the barcode can remain intact for decades or even hundreds of years in some cases. This barcode longevity can be achieved using DNA data storage. More precisely, the information regarding the product can be encoded into DNA and stored into the concerned material to be retrieved in the long-term without loss of information. Regarding the storage of the DNA into the raw materials there are multiple solutions that can be used according to the different use cases of this application. The DNA containing the barcode needs to be protected from contacts with water and oxygen to ensure reliability of the molecule. To this end, it can be either inserted into sealed capsules [33] which can ensure reliability for 170.000 years in room temperature or another option for example is the DNA encapsulation into silica glass beads [34] that can protect DNA for 20-90 years in room temperature and can go up to 2000 years if kept in 9.4 degrees Celsius. The encapsulated DNA is then integrated into the different materials to signify the authenticity and the list of ingredients of the referring material.

6. DNA-based Media Storage: Requirements

Although this is still rather preliminary, the potential list of requirements may include:

1. **Compression efficiency -** The standard **shall** offer significantly increased compression efficiency regarding simple solutions in the literature, e.g. based on binary transcoding. This includes various media modalities including those where redundancy has to be exploited across multiple components such as

volumetric images.

2. **Lossless coding -** The standard **shall** offer lossless coding at a state-of-the art compression performance.

3. **Composite media assets -** The standard **shall** offer the capability to represent composite sets of multiple, related elementary media assets, notably images (and associated metadata), e.g. by means of appropriate auxiliary data.

4. **Metadata -** The standard **shall** offer the capability to efficiently code relevant metadata, associated to the relevant media assets, available or not in a binary representation.

5. **Privacy and Security** – The standard **shall** offer the capability to efficiently code appropriate privacy and security related data, associated to the relevant media assets.

6. **Random access -** The standard **shall** allow the access to specific parts of the information without having to decode the full coded information.

7. **Biological constraints -** The standard **shall** consider the relevant biological constraints on the coding process to avoid affecting the stability of the sequence and synthesising and sequencing errors, e.g. avoiding long homopolymers (repeats of the same nucleotides > 3) and extreme G-C content.

8. **Error resilience -** The standard **shall** offer some degree of error resilience regarding reading/sequencing errors, including unequal protection against errors and those taking into account the special nature of errors in DNA-based storage

9. **Scalability -** The standard **shall** allow scalable representations of the information where reading only part of the full information offers a lower quality or resolution of the full represented information.

10. **Ambiguity -** The standard **shall** allow decoding without any ambiguity, i.e. a decoded bit may not be both '0' and'1'.

11. **Artificial recognition -** The standard **shall** allow the encoding output to be unambiguously recognized as artificial DNA; this is relevant as the artificial DNA stream should not be confused with natural DNA streams.

12. **Biosafety -** The standard **shall** not allow encoding outputs which constitute any danger in terms of biosafety. The standard **shall** define mechanisms and conditions to ensure biosafety.

7. Relevant DNA-based Media Storage Companies, Initiatives and Consortia

Researchers at the University of Washington and Microsoft Research have developed a fully automated end-to-end system for writing, storing and reading data encoded in DNA [4]. According to [4], *"Microsoft is exploring ways to close a looming gap between the amount of data we are producing that needs to be preserved and our capacity to store it. That includes developing algorithms and molecular computing technologies to encode and retrieve data in fabricated DNA, which could fit all the information currently stored in a warehouse-sized data center into a space roughly the size of a few board game dice."*

A number of companies, including Microsoft and Twist Bioscience, are working to advance DNA-storage technology [1].

A consortium, named *DNA Data Storage Alliance,* is being created to define an interoperable end-to-end architecture for data storage based on DNA and to accelerate the creation of an ecosystem. The DNA Data Storage Alliance will be a global ecosystem of companies and academic researchers, setting industry-leading DNA data storage software and hardware standards and specifications that enable and streamline the use of DNA to store digital data.

Oligoarchive [27], Intelligent DNA Storage for Archival, is a European Commission funded FET project that aims at defining an architecture with the same name for efficient DNA storage of digital information.

8. What Role for JPEG and Next Steps

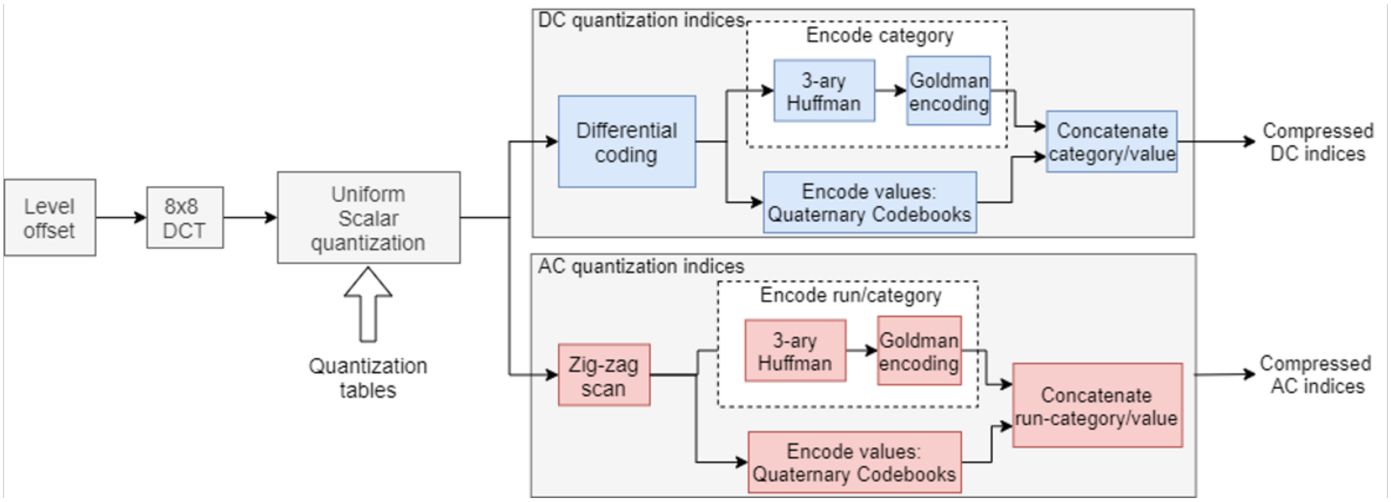
Because of its past successful history of offering efficient image and image sequence formats for storage and archival applications, the JPEG committee is well positioned to address standardization challenges related to multimedia content efficient representations and, in particular, for image and image sequences in the context of DNA storage.

As a minimum, the JPEG committee can launch an activity to convert its existing image coding formats from compressed binary representation to compressed DNA 4-ary representation according to appropriate requirements. Standardized image coding solutions along with complementary appropriate tools, such as error resiliency and associated metadata, which particularly suit the requirements of DNA digital information storage, are good directions for JPEG to explore.

As a next step, the applications of DNA digital information storage need to be explored more in detail with particular emphasis on image and video content as information. They should then be ordered in terms of time to market and maturity and efforts should be focused on a specific use case that can gather a critical mass of stakeholders while remaining open to other use cases.

9. JPEG DNA experimentation software

The JPEG DNA python package is an implementation of the algorithm of the same name. The main purpose of the JPEG DNA algorithm is to encode the DCT coefficients into DNA-like quaternary code instead of binary code. This work has been published in [42]. First, a quantization is applied, then a zigzag transform, the result is a sequence of integers in which the first element is the DC coefficient, then follow all the 63 AC coefficients.



The encoding system for DC and for AC coefficients share similarities with the usual JPEG algorithm but have their own specificity, especially in the coding of each value. Like in the JPEG algorithm, we use a system of categories and run/categories. The categories are used for the DC values and describe the interval in which the value falls. The run/categories in addition to the category of the coefficient also groups the number of zeros before this value.

The difference with the usual JPEG algorithm lies in the way categories, run/categories and values are coded. They are not anymore coded into binary data but into quaternary DNA-like data. This data will be synthesized into DNA molecules and the data content will be sequenced (read) from those molecules. The technologies are restricting the possibilities of quaternary sequences that can be encoded: we can’t have homopolymers (repetitions of the same base), and the bases use must be balanced (G and C base must represent between 40% and 60% of the totality of the content).

To encode the categories and run categories, we use a combination of ternary Huffman coding with Goldman coding [6] on the [“A”, “T”, “C”, “G”]. The Huffman coder generates codewords recursively with the alphabet {0,1,2} according to their frequency. The most frequent categories have a shorter codeword. The Goldman coder translates the ternary bases of these codewords with a rotating {A, T, C, G} alphabet into quaternary DNA-like codewords.

The values are coded using fixed-length codebooks that have been pre-generated and that respect the constraints previously described. We concatenate all the run/category and value codes to obtain a stream for the block. We concatenate the block codes to obtain the stream for the image.

The development of this algorithm has been oriented towards an easy-to-use, easy-to-modify paradigm. As previously described, the algorithm is working with a collection of coders (Huffman, Goldman), transforms (DCT, Zigzag). The library is composed of three main packages, one for coders, one for transforms and one for codecs. The latter one only contains the JPEG DNA codec. The coders and transforms used by these codecs are developed as modules in the packages of the same name. Also in the scripts package, one will find and may add general scripts for those codecs to perform compression, decompression, performance evaluation, etc...

A thorough explanation of the package can be found in the input document number M93103, and a wiki for documentation and link to the code can be found here www.i3s.unice.fr/~am/JpegDNA/. In order to access the code please contact xpic@i3s.unice.fr.

10. JPEG DNA exploration experiments

In this section we identify a number of potential exploration experiments that are proposed to be the focus of concrete collaborative activities in order to produce a common test condition document and accompanying software to pursue this activity in view of creating an anchor reference implementation with good efficiency as well as in realistic conditions in identified use cases.
The list of potential exploration experiments are as follows:

- **Exploration experiments to include overhead in JPEG DNA compressed quaternary streams** (I3S, EPFL).
  - The format of the encoded quaternary stream must be defined (considering the size of the oligos)
  - Inclusion of JPEG DNA stream of necessary metadata for decoding purposes and definition of the format of such metadata
- **Exploration experiments on solutions to make JPEG DNA quaternary streams more robust** (I3S, EPFL).
  - Propose efficient consensus methods that provide best results in noisy environments
  - Define quaternary error correction codes
  - Cope with EOB error propagation
- **Exploration experiments for realistic and effective error simulations** (I3S, EPFL).
  - Deep Simulator
  - MESA
  - Imperial College
- **Exploration experiments for the definition of common test conditions** (CTC) (EPFL, I3S, UBI).
  - Data set to use
  - Parameters to use
  - Assessment methodologies to use

11. Dissemination and Participation

To collect information on DNA-based media storage technologies as well as target use cases and requirements, JPEG is organising a series of workshops. The workshops and discussion sessions are organized with experts and end users in order to better understand the market needs and how a JPEG standard can help create or accelerate an ecosystem for media storage on DNA. Once the latter is identified, the standardization process can start with precise milestones to be identified for each stage.

The first JPEG DNA workshop happened on September 30, 2020, with the following program:

4:00pm UTC: Introduction, Touradj Ebrahimi (EPFL)

4:05pm UTC; DNA-based Media Storage: Exploring New Frontiers, Fernando Pereira (IST-IT)

4:25pm UTC: Image coding for long-term storage on synthetic DNA, Marc Antonini (Université Côte d'Azur, CNRS, I3S)

4:55pm UTC: Microform and Macromolecules: Issues in archiving digital data on analog or biological storage media, Raja Appuswamy (EURECOM)

5:15pm UTC: Wrap up session

5:30pm UTC: End

The second JPEG DNA workshop happened on January 8, 2021, with the following program:

8:00 UTC - Introduction, Touradj Ebrahimi (EPFL) and Fernando Pereira (IST-IT)

8:05 UTC - An overview of DNA Data storage alliance, Daniel Chadash (Twist Bioscience)

8:25 UTC - OligoArchive - Using Synthetic DNA to Store and Process Data, Thomas Heinis (Imperial College)

8:45 UTC - Montreux Jazz Digital Project, from a heritage to an innovation platform, Alain Dufaux (EPFL)

9:05 UTC - Wrap up session

9:30 UTC - End

The third JPEG DNA workshop happened on April 9, 2021, with the following program:

17:00 UTC - Introduction, Touradj Ebrahimi and Fernando Pereira

17:05 UTC - Audiovisual Digital Preservation and DNA Storage Requirements, Linda Tadic (Digital Bedrock and UCLA Department of Information Studies)

17:25 UTC - Low cost DNA data storage with noisy synthesis and advanced error correction, Reinhard Heckel (Technical University of Munich and Rice University)

17:45 UTC - Coding for efficient DNA data storage, Sergey Yekhanin (Microsoft Research)

18:05 UTC - Wrap up session

18:30 UTC - End

The fourth JPEG DNA workshop happened on July 2nd, 2021, with the following program:

15:00 UTC - Introduction, Touradj Ebrahimi and Marc Antonini

15:05 UTC - Compression, learning and error-correction for DNA-based image storage, Olgica Milenkovic (University of Illinois, Urbana-Champaign, USA)

15:30 UTC - Writing data to DNA: What breakthroughs are needed to make this technology feasible?, Nimesh Pinnamaneni (HelixWorks Technologies, Ireland)

15:55 UTC - Imagene technology for ambient temperature storage of DNA, Marthe Colotte (Imagene, France)

16:20 UTC - Wrap up session

16:45 UTC - End

The fifth JPEG DNA workshop happened on October 8, 2021, with the following program:

15:00 UTC - Introduction, Touradj Ebrahimi and Marc Antonini

15:05 UTC - DNA synthesis for data storage: a perspective for information theorist, Anthony Genot (LIMMS/CNRS, Japan)

15:35 UTC - Efficient Synthesis of DNA via Cost-Constrained Systems, Andreas Lenz (Technical University of Munich, Germany)

16:05 UTC - Wrap up session

16:30 UTC - End

The presentations and recordings from these workshops are publicly available at the JPEG website, https://jpeg.org/.

Naturally, the most important JPEG DNA dissemination document is this document itself, which most recent version may be always found at https://jpeg.org/jpegdna/documentation.html.

To participate in the discussions related to the JPEG DNA activity, experts are invited to subscribe to the mailing list at http://listregistration.jpeg.org.

For further information please see [28], [29],[30],[31],[32] for some videos of DNA-based media storage.

References

1. *”DNA data storage is closer than you think”, https://www.scientificamerican.com/article/dna-data-storage-is-closer-than-you-think/.*

2. *“DNA”, https://en.wikipedia.org/wiki/DNA#Nucleobase_classification.*

3. *http://www.core.org.cn/NR/rdonlyres/Biology/7-A12Fall-2005/D4134A30-F348-4615-8B0A-D0CB5ED86081/0/chp_dna.jpg.*

4. *“With a “hello”, Microsoft and UW demonstrate first fully automated DNA data storage”, https://news.microsoft.com/innovation-stories/hello-data-dna-storage/.*

5. *M. Dimopoulou, M. Antonini, P. Barbry, R. Appuswamy, “DNA Coding for image storage using image compression techniques”, CORESA, Poitiers, France, Nov. 2018.*

6. *N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E.M. LeProust, B. Sipos, and E. Birney, “Towards practical, high-capacity, low-maintenance information storage in synthesized DNA”, Nature, vol. 494, no. 7435, pp. 77–80, 2013.*

7. *https://www.nonteek.com/en/dna-storage-solution-to-big-data-in-a-strand/.*

8. *G. M. Church, Y. Gao, and S. Kosuri, “Next-generation digital information storage in DNA,” Science, vol. 337, no. 6102, pp. 1628–1628, Aug. 2012.*

9. *C. Bancroft, “Long-term storage of information in DNA,” Science, vol. 293, no. 5536, pp. 1763c–1765, Sep. 2001.*

10. *M. Campbell, "DNA data storage: automated DNA synthesis and sequencing are key to unlocking virtually unlimited data storage," Computer, vol. 53, no. 04, pp. 63-67, 2020.doi: 10.1109/MC.2020.2967908.*

11. *R. N. Grass, R. Heckel, M. Puddu, D. Paunescu, and W. J. Stark, “Robust chemical preservation of digital information on DNA in silica with error-correcting codes,” Angewandte Chemie International Edition, vol. 54, no. 8, pp. 2552–2555, 2015.*

12. *M. Blawat, K. Gaedke, I. Huetter, X.-M. Chen, B. Turczyk, S. Inverso, B. W. Pruitt, and G. M. Church, "Forward error correction for DNA data storage," Procedia Computer Science, vol. 80, pp. 1011-1022, 2016.*

13. *S. H. T. Yazdi, Y. Yuan, J. Ma, H. Zhao, and O. Milenkovic, "A rewritable, random-access DNA-based storage system," Scientific reports, vol. 5, p. 14138, 2015.*

14. *Y. Erlich and D. Zielinski, "DNA fountain enables a robust and efficient storage architecture," Science, vol. 355, no. 6328, pp. 950-954, 2017.*

15. *J. Bornholt, R. Lopez, D. M. Carmean, L. Ceze, G. Seelig, and K. Strauss, "A DNA-based archival storage system," ACM SIGOPS Operating Systems Review, vol. 50, no. 2, pp. 637-649, 2016.*

16. *L. Organick, S. D. Ang, Y.-J. Chen, R. Lopez, S. Yekhanin, K. Makarychev, M. Z. Racz, G. Kamath, P. Gopalan, B. Nguyen et al., "Scaling up DNA data storage and random access retrieval," bioRxiv, p. 114553, 2017.*

17. *C. Rashtchian, K. Makarychev, M. Racz, S. Ang, D. Jevdjic, S. Yekhanin, L. Ceze, and K. Strauss, "Clustering billions of reads for DNA data storage," in Advances in Neural Information Processing Systems, 2017, pp. 3362-3373.*

18. *M. Dimopoulou, E. Gil San Antonio and M. Antonini, "An Efficient Sequencing Noise Resistant Mapping for the Encoding of Images onto Synthetic DNA, IEEE Multimedia Signal Processing (MMSP), Sept. 2020.*

19. *M. Dimopoulou, M. Antonini, P. Barbry and R. Appuswamy, "A biologically constrained encoding solution for long-term storage of images onto synthetic DNA," European Signal Processing Conference (EUSIPCO), Sept. 2019.*

20. *M. Dimopoulou, M. Antonini, P. Barbry and R. Appuswamy, "Storing Digital Data Into DNA: A Comparative Study Of Quaternary Code Construction," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, May 2020.*

21. *M. Dimopoulou and M. Antonini, "Efficient Storage of Images onto DNA Using Vector Quantization", Data Compression Conference (DCC), Mar. 2020.*

22. *M. Dimopoulou and M. Antonini, "Image storage in DNA using Vector Quantization", European Signal Processing Conference (EUSIPCO), Aug. 2020.*

23. *L. Anavy, I. Vaknin, O. Atar et al. Data storage in DNA with fewer synthesis cycles using composite DNA letters. Nat Biotechnol 37, 1229–1236 (2019). https://doi.org/10.1038/s41587-019-0240-x.*

24. *"Deep Purple's "Smoke on the Water" Becomes a Piece of Scientific History, https://www.twistbioscience.com/blog/company-news-updates/deep-purples-smoke-water-becomes-piece-scientific-history.*

25. *N. Goela and J. Bolot, "Encoding movies and data in DNA storage," 2016 Information Theory and Applications Workshop (ITA), La Jolla, CA, US, 2016, pp. 1-1, doi: 10.1109/ITA.2016.7888163.*

26. *"Making DNA data storage a reality", https://www.the-scientist.com/cover-story/making-dna-data-storage-a-reality-30218.*

27. *Oligoarchive, https://oligoarchive.github.io.*

28. *"DNA data storage is the future!", https://www.youtube.com/watch?v=aPWA-n9oo4k.*

29. N. Goldman, "DNA Hard Drives", https://www.youtube.com/watch?v=tBvd7OSDGgQ.

30. D. Zielinski, "How we can store digital data in DNA", https://www.youtube.com/watch?v=wxStlzunxCw.

31. "Microsoft and UW demonstrate first fully automated DNA data storage", https://www.youtube.com/watch?v=60Gi5lqL-dA.

32. "Storing data in DNA", https://www.youtube.com/watch?v=vjc4LIcoux4.

33. Kevin Washetine, Simon Heeke et al. "DNAshell Protects DNA Stored at Room Temperature for Downstream Next-Generation Sequencing Studies" Biopreservation and Biobanking 2019 doi.org/10.1089/bio.2018.0129

34. Paunescu, D., Puddu, M., Soellner, J. et al. Reversible DNA encapsulation in silica to produce ROS-resistant and heat-resistant synthetic DNA 'fossils'. Nat Protoc 8, 2440–2448 (2013). https://doi.org/10.1038/nprot.2013.154

35. Heckel, R., Mikutis, G. and Grass, R.N. A Characterization of the DNA Data Storage Channel. Sci Rep 9, 9663 (2019). https://doi.org/10.1038/s41598-019-45832-6

36. Matange, K., Tuck, J.M. and Keung, A.J. DNA stability: a central design consideration for DNA data storage systems. Nat Commun 12, 1358 (2021). https://doi.org/10.1038/s41467-021-21587-5

37. C. Yang, J. Chu, R. L. Warren, and I. Birol, "Nanosim: nanopore sequence read simulator based on statistical characterization, "GigaScience, vol. 6, no. 4, p. gix010, 2017.

38. Y. Li, R. Han, C. Bi, M. Li, S. Wang and X. Gao, "DeepSimulator: a deep simulator for Nanopore sequencing", Bioinformatics, Volume 34, Issue 17, 01 September 2018, Pages 2899–2908, https://doi.org/10.1093/bioinformatics/bty223

39. Y. Li, S. Wang, C. Bi, Z. Qiu, M. Li, and X. Gao, "Deepsimulator1. 5: a more powerful, quicker and lighter simulator for nanopore sequencing," Bioinformatics, vol. 36, no. 8, pp. 2578–2580, 2020.

40. M. Schwarz, M. Welzel, T. Kabdullayeva, A. Becker, B. Freisleben, D. Heider, "MESA: automated assessment of synthetic DNA fragments and simulation of DNA synthesis, storage, sequencing and PCR errors", Bioinformatics, Volume 36, Issue 11, June 2020, Pages 3322–3326, https://doi.org/10.1093/bioinformatics/btaa140

41. S. Alosaimi, A. Bandiang, N. van Biljon, et al, "A broad survey of DNA sequence data simulation tools", Briefings in Functional Genomics, Volume 19, Issue 1, January 2020, Pages 49–59, https://doi.org/10.1093/bfgp/elz033

42. M. Dimopoulou, E. Gil San Antonio, M. Antonini, "A JPEG-based image coding solution for data storage on DNA", European Signal Processing Conference (EUSIPCO), Aug. 2021.