Synthetic Biology: Secure Digital Storage, DNA-based Computation and the Organic Computer

Alexander R Widdel
Division of Science and Mathematics
University of Minnesota, Morris
Morris, Minnesota, USA 56267
widd0014@morris.umn.edu

ABSTRACT

This paper addresses the role of synthetic biology in computer science. Some applications of synthetic biology in computing include the storage of digital information in organic material, cryptographic security via DNA-based encryption, and biological computation. Furthermore, development of a universal operating system for the biological cell, meaning to simplify the process of coding for organic systems or bio-computers is being attempted by the University of Nottingham's AUdACiOuS project. Though meaningful progress has been made in the field, much of the work of synthetic biology is still precursory.

1. INTRODUCTION

In the future, the hardware-software paradigm as we know it may be complemented, and perhaps in some places substituted, by biological alternatives. Advancement in synthetic biology can only be accomplished through collaborative work between biologists and computer scientists. Mutual comprehension of the dynamic behavior of cells is also required. Throughout this paper, the practical uses of cellular or molecular functions and procedures in storage, information security, computation and the construction of biocomputers are addressed. Molecules known as DNA and RNA serve as the basis for the majority of these processes. The advantages, drawbacks and future prospects of the use of DNA for these processes will be discussed. A traditional computer must perform operations like reading or writing bits and executing logical and arithmetic operations. DNA can encode bits in a way that is compatible with the way computers store information. The equivalent to writing bits in a biological system is DNA synthesis, while the equivalent of reading bits is DNA sequencing.

Computation with biological components will be addressed in greater detail in this paper. Background will first be provided for the biological, cryptographic, and systems material. Section 3 provides a comparison between traditional and DNA-based storage. Section 4 addresses the potential uses of DNA to securely encrypt data. Finally, section 5 will focus primarily on computing with biological components.

UMM CSci Senior Seminar Conference, December 2014 Morris, MN.

2. BACKGROUND

To understand the fundamentals of using biology as a technology, a basic comprehension of the structure and behavior of Deoxyribonucleic acid (DNA) and ribonucleic acid (RNA) is required. It is also important to note the distinction between genetic modification and the practice of synthetic biology. Genetic modification - splicing alternate genes into living embryos to produce organisms with novel properties - is performed *in vivo*. In vivo operations are performed within and between complete and living organisms. This differs from the practice of synthesizing DNA *in vitro* (in a controlled environment outside of a living organism) for use in living organisms or as a biological component [12]. This section will also cover necessary background in cryptographic methods and systems architecture.

2.1 Biology

DNA is a complex molecule comprised of four elementary units that can be seen as "encoding" the information contained within. Structurally, it resembles a "double-helix," with two strands of these units (called nucleotides, or bases) linking together and spiraling around one another. The four bases are Adenine (A), Thymine (T), Cytosine (C) and Guanine (G). A links with T and C links with G in the spiral. Ribonucleic acid or RNA is very similar to DNA, differing in that it is commonly single-strand, and the base Uracil replaces Thymine, binding to Adenine. DNA can also exist as a single strand. The main role of DNA molecules is the long-term storage of information. Unlike typical electronic storage, data that is deleted cannot be reread [6]. In order to create RNA, DNA undergoes transcription.

2.1.1 Transcription

During transcription, a part of the cell's "machinery" called RNA polymerase moves along the DNA strand. The apparatus unzips the two halves of the strand, and reads one of them, one base at a time, as it progresses. As it reads the DNA, it adds the opposite base (also supplanting Uracil for Thymine) to a "messenger" RNA (mRNA) strand. This mRNA strand with opposing bases is said to be *complementary* to the original template DNA. In order to create a protein, mRNA undergoes translation.

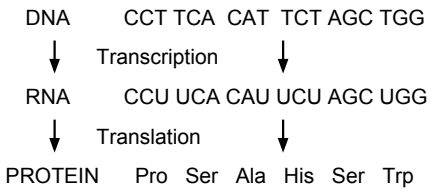


Figure 1: Information Flow: DNA to RNA to Proteins

2.1.2 Translation

In the process of translation, mRNA travels to another molecular machine known as a "ribosome" which reads its bases in groups of three. These base triplets - called codons - determine the addition of one of 20 protein subunits called amino acids to a chain that is crafted by the ribosome. A completed chain of these amino acids constitutes a complete protein. Once formed, the protein undergoes a folding process, forming itself into a shape which will define its function. Proteins are responsible for nearly all chemical reactions within cells. The information flow in biological systems of "DNA to RNA to proteins" is known as the *central dogma of genetics*.

2.1.3 Copying DNA

Duplication - also called amplification - of DNA is accomplished using polymerase chain reaction, or PCR. This process is distinct from the operation of copying data on a hard drive, as potentially billions of copies of the original DNA are being generated. In PCR, DNA is heated in order to be denatured or "unzipped" into two strands. Then, relatively short "oligos" which match the beginning and end of the DNA template are attached to their matching sequence. An oligo is a short strand of DNA. These oligos are called primers. Finally, after being heated again, a cellular apparatus called DNA polymerase binds to each primer and works its way along its strand, adding complementary bases until replication is complete. The newly formed strands are then denatured and replicated repeatedly, yielding roughly one million copies after 20 cycles, and one billion after 30. PCR takes approximately 2 hours to complete. Approximately 1 in 10,000 bases are duplicated incorrectly per cycle. Errors which occur early in the process will accumulate, negatively impacting its fidelity. For comparison, data transfer in a typical 7200 RPM desktop HDD, as of 2010, rates up to 1030 Mbit/s.

2.2 The One Time Pad

Cryptography is concerned with the secure encryption of data into a form unreadable to any party without access to the key which decrypts it. Encryption is the process of scrambling the original message (plain-text) message into an encrypted message (cipher-text). One method of encryption is called the One Time Pad or OTP. First, a long sequence of random bits are generated. The pad must be of greater or equal length to the data it is encrypting. Each of these bits is *exclusive-or*'ed with the plain-text to create the cipher-text. The exclusive-or or "$\oplus$" operator can be viewed as addition modulo 2, as seen in table 2.

A	B	A $\oplus$ B
0	0	0
0	1	1
1	0	1
1	1	0

Table 1: $\oplus$ Truth Table

Consider plaintext A, OTP B, and cipher-text C.

$$C = A \oplus B$$

Notice that the cipher-text can be inverted again to retrieve the original plaintext:

$$C \oplus B = A$$

The resulting cipher-text will have a uniform frequency distribution, meaning that it is just as likely as any of the other possible 26^n encipherings. The pad is shared secretly with the recipient, and is XOR'ed again with the cipher-text to decrypt the message. It has been mathematically proven that the OTP produces a theoretically unbreakable cryptosystem for encrypting and decrypting data [17]. The OTP achieves perfect secrecy in the sense that it is information-theoretically secure, meaning that the encrypted message or cipher-text provides no information about the original plaintext. The security of OTP relies on the pad being discarded after use. DNA is a favorable medium for application of the OTP as a small amount of DNA can store enormous pads.

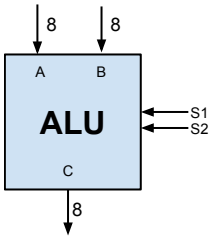
2.3 Systems

This section will review the basic classical computer architecture. Applications of systemics to synthetic biology have led to the conception of silicon computers and their engineering as a blueprint for the development of a similar apparatus made up of biological components. The current model of the bio-computer follows the Von Neumann computer architecture with four units: an input/output device, an arithmetic logic unit, a control unit and wires (bus) to interconnect these components. Such a bio-computer could operate within a living organism, observe its environment and serve to control biological systems [13].

A simple arithmetic logic unit (as seen in Figure 2) can take in two control bits, S_1 and S_2, which can encode four operations for the unit to perform. The input S_1S_2 will determine which operation will be performed on two 8-bit inputs. Consider control bits $S_1S_2 = 10$ and input bits 10111010 and 00010110. Control bits 10 code for the *not* operation, which will consider only the first input bitstream. The not operation simply flips every bit in the bitstream. Here, the final output would be 01000101. This ALU would be sufficient for performing addition and subtraction on integers. Section 5 will include details on building a biological ALU.

3. TRADITIONAL VS ORGANIC STORAGE

A lot of attention has been drawn in recent research efforts to next-generation storage media [5][6]. Efficient long term and high capacity digital storage is becoming more important as the world's storage needs curve sharply upwards. Hard drive storage is increasingly cheaper, faster, and more dense. Costs for hard disk drives have been dropping at a rate of 1.6-fold per year. By contrast, the cost of DNA synthesis and sequencing have been dropping at rates of 5 and



S1 S2	Operation
0 0	C = A + B
0 1	C = A + 1
1 0	C = NOT A
1 1	C = 0

Figure 2: Arithmetic Logic Unit

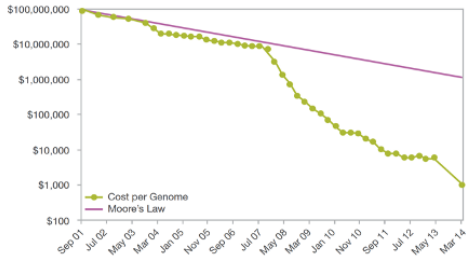


Figure 3: Genome Cost vs Moore's Law [8]

12-fold per year respectively [5]. Comparisons have been drawn between the rate of dropping costs in DNA sequencing and the benchmark of Moore's Law, as shown in figure 3.

It is important to note that prices drop in fits and starts rather than in a consistent exponential fashion. However, the coveted $1000 sequencing of the human genome is now promised by the Illumina Corporation (revealed on January 14, 2014 to be owned by Apple), making large-scale genome sequencing more affordable than ever.

The maximum capacity of hard disk drives has been pushed to 8 terabytes by Seagate [15]. More compact means of data storage such as microSD cards have capacities up to 128 gigabytes, but synthetic biology explores a medium that is truly ultra compact: that of DNA. This is a promising medium due to its tremendous storage capacity in comparison to physical space, with a storage density of 5.5 petabits per mm^3 [5]. It greatly exceeds the capacity of electronic, magnetic and optical media. A gram of single-stranded DNA has been shown to hold up to a theoretical maximum of 455 exabytes of raw data. An exabyte is approximately 1024 petabytes, or 1 billion gigabytes.

Current encoding schemes allow for an arbitrary amount of data by separating a long DNA strand into blocks. The process of synthesis begins with designing and synthesizing oligos with index tags (fixed-length strings of DNA noting each oligo's position of the gene) and assembling them in order to obtain full-length strands. Then, error correction handles mutations and the final sequence is re-confirmed for accuracy. The number of bases needed to encode data grows linearly with the amount of information to be stored, but the index bits required to assemble full-length files from short fragments must also be accounted for. Since index bits have logarithmic growth with the number of fragments to be indexed, the total amount of synthesized DNA required grows sub-linearly [6]. Along with its dense capacity, DNA has the advantage of longevity, able to be read after long periods if kept under relatively easily achieved conditions [6].

3.1 DNA Synthesis (Writing)

There are six steps in the build cycle of a "laser printing" DNA synthesis method offered by a startup called Cambrian Genomics:

1. DNA chip synthesis
2. DNA released
3. DNA captured on microbeads
4. Beads make roughly 100,000 copies of DNA (PCR)
5. Beads are sequenced for quality control
6. Laser-pulse catapulting of valid DNA

A DNA chip or biochip is a collection of microscopic indexed wells containing DNA strands adhered to a glass, plastic or silicon surface known as a genome chip or gene array. In this process, the beads with multitudes of copied DNA strands are sequenced and the accurately synthesized strands are catapulted via laser from the slide onto a collector plate. This method can print up to a hundred strands per second.

Writing data into synthesized DNA entails the assignment of the A or C bases as ones and the Gs or Ts as zeroes. Some applications encode one bit per base, while others encode two. Encoding one bit per base allows for more variation in the bit representation of data. This allows problematic structures like long sections of repeated bases or palindromic sequences to be avoided. Certain methods have higher error rates for sequencing palindromic strands.

Below is an example of a 2 bit per base scheme of encoding binary information into DNA:

Binary Sequence	Base
00	T
01	G
10	C
11	A

The binary string 01001000 01100101 01101100 01101100 01101111 00100000 01010111 01101111 01110010 01101100 01100100 (Hello World) can therefore be expressed as GTCTGCGGGCATGCATGCAATCTTGGGAGCAAGATCGCATGCG.

3.1.1 Unnatural Base Pairs

Additionally, synthesized DNA strands have the potential to include bases not found in nature, known as unnatural base pairs (UBP). Two synthetic bases - d5SICSTP and dNaMTP or X and Y for short - have been incorporated into a partially-synthetic E.coli which could successfully self-replicate, making for a proof-of concept for UBP [12]. So far these bases have not been capable of making proteins. However, the expansion of the genetic code presents an opportunity for alternative encoding schemes.

Phred Quality Score	Probability of Incorrect Base	Base Accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1,000	99.9%
40	1 in 10,000	99.99%
50	1 in 100,000	99.999%
60	1 in 1,000,000	99.9999%

Table 2: Phred Error Probabilities

3.2 Reading (Sequencing)

Reading data that is written into DNA is done by sequencing the encoded chain of bases. The accuracy of each sequenced base is decided by the Phred quality score metric. Quality score $Q = -10\log_{10} P$ where P refers to the probability of a base being labeled incorrectly.

Sequencing DNA in vivo was the primary bottleneck for accelerating the process. In vitro techniques were thus adopted to eliminate the need for manual processing, and improve scalability and automation. The first hand-held, single-molecule DNA sequencer has been developed by New Zealand scientists and should simplify reading DNA-encoded data going forward.

One particularly cheap and high-throughput (population-scale) method of sequencing, provided by Illumina, is called sequencing by synthesis. This method sequences 50 to 300 base pairs for $.05-$.15 per base pair, but can take up to ten days depending upon the sequencer. The initial capital cost of the machinery is quite high and large concentrations of DNA are required for its use. The Illumina HiSeq X Ten sequencer (actually 10 HiSeq X machines run in parallel) costs $10 million, and an optional $1 million for every additional HiSeq sequencer. Illumina promises that for the HiSeq X sequencer, 95.2% of bases have a quality score over 30.

3.2.1 Processing

Sequencing by synthesis is performed through a process of dye sequencing. It begins with the attachment of DNA molecules to a DNA chip, followed by amplification of that DNA to produce local colonies. Then, a cluster of all four bases is added, each fluorescently labeled with a different color and attached with an index tag. The introduced bases compete for binding sites on the chip and the molecules that do not bind are discarded. A laser is then used to excite the dyes and a photograph of the bases that do bind is taken. This process repeats until the molecule is completely sequenced.

3.3 Results

Recent work has successfully encoded a 5.27 megabit bitstream of text, jpg images and a javascript program. The bits were encoded into 54,898 159-base oligos, each encoding a 96 bit data block. Each block includes a 19 bit address specifying the location of the block in the bit stream. All data blocks were recovered with a total of 10 bit errors, which were predominantly located within blocks of repeated bases [5]. This makes for a bit error rate of 2.37E-7, compared to a typical HDD write error rate of 1.0E-14. Writing this data takes days, and synthesis is costly, making DNA storage primarily an archiving tool for data with an access rate as infrequent as a century, but write time is improving. DNA synthesis costs are currently being reduced at a pace that ought to make it cost-effective for sub-50-year archiving within a decade [6].

4. CRYPTOGRAPHIC METHODOLOGIES IN ORGANIC STORAGE

The use of DNA as a storage medium also has implications in data encryption. For instance, DNA is a favorable medium for implementation of the OTP, as the pad for a long message can be impractically large. This is a lesser concern given the storage density of DNA, which allows for a very long sequence to be shared easily with only one exchange rather than several. Despite this advantage, there are at least two important trade-offs. There are multiple copies of the DNA, making it possible for DNA to be shed for a waiting attacker to harvest, and sequencing can introduce error. A concrete example of this cryptosystem is the encryption tool DNACrypt, available for free online. DNACrypt initially generates a secret library of one-time pads in the form of strands of DNA. This library is known as a "codebook." There are two pre-requisites for a codebook to be considered secure. Their contents must be truly random, and they must be used only once. Then, an algorithm may be applied to further encrypt the message. The pads are used to encrypt plain-text via XOR computation with a DNA chip. For the purposes of this cryptosystem it is assumed the plain-text is encoded in DNA strands.

4.1 Encryption via XOR operations on DNA Chips

Once the plaintext encoded into synthesized bases is created, DNACrypt applies the OTP as follows: Let L be the number of bits S that remain unused. When a plain-text binary message M which is n < L bits long needs to be sent, each bit M_i is XOR'ed with the bit $K_i = S_R$ - L + i to produce encrypted bits C_i=$M_i \oplus K_i$ for i = 1,...,n. The n bits of S that have been consumed are then destroyed at the source and the encrypted sequence. The identical process is repeated at the message's destination, where the encrypted sequence is used in the place of M. This reproduces the initial message since $C_i \oplus K_i = M_i$ since $C_i = M_i \oplus K_i$ and $M_i \oplus K_i \oplus K_i = M$. Each of the one-time pad DNA sequences are also assumed to have appended unique prefix index tags of a fixed length, which form the complements of the plain-text message tags. Each corresponding pair of a plain-text message and a one-time-pad sequence, with the same tag, can be concatenated into a single DNA strand.

5. BIOLOGICAL COMPUTATION

Data processing is seen in nature in many forms: DNA information storage, intra- and extra-cellular communication, and systems such as the immune system and the nerve system can be depicted abstractly as computational systems. Complex problems have been solved with programming paradigms inspired by natural processes. Artificial neural networks, evolutionary algorithms, genetic programming, artificial immune systems and cellular automata have all been born of natural systems. These methods differ from the use of natural materials for computing, but demonstrate that biologically-inspired programming can be effective.

5.1 Original Bio-Computer

The first bio-computer based on DNA was built in 1994 by Leonard Adleman, and the system was capable of solving a seven node instance of the directed Hamiltonian path problem [13]. A Hamiltonian path is a path that moves through every vertex of a graph, touching each node exactly once. A directed graph was encoded in molecules of DNA, and the operations of the computation were performed by enzymes. Solving a seven node instance is a trivial computation, however it provides a proof of concept for DNA computing. Leonard's concept was an analog computing framework disparate from the digital binary system described earlier, with the graph's structure encoded more directly by the DNA. As seen in table 3, for each vertex i in the graph, a random 20-base oligo O_i is generated. For edge $i \rightarrow j$ in the graph, an oligo $O_{i \rightarrow j}$ is derived from O_i and O_j. For each vertex i in the graph, $\bar{O}_i$ is the complement of O_i. Here, $\bar{O}_3$ serves as a splint to bind $O_{2 \rightarrow 3}$ and $O_{3 \rightarrow 4}$. This approach takes advantage of parallel computing in that many molecules of DNA may try many different possibilities at once in order to solve problems. For the Hamiltonian path problem, this means generate random paths through the graph and discarding those that don't conform to the properties of a valid path. If any paths remain, the output is true (indicating that a Hamiltonian path was found), otherwise false. Currently, DNA computers are only faster than their silicon equivalents in certain specialized problems (like the assignment problem [16] and Strassen's matrix multiplication algorithm [14]). The great potential computing power of bio-computers believed by Adleman has yet to be empirically demonstrated. Adleman's DNA graph was also not a universal computer. In order to achieve a general Von Neumann computer one would need the components discussed earlier.

O_2	TATCGGATCGGTATATCCGA
O_3	GCTATTCGAGCTTAAAGCTA
O_4	GGCTAGGTACCAGCATGCTT
$O_{2 \rightarrow 3}$	GTATATCCGAGCTATTCGAG
$O_{3 \rightarrow 4}$	CTTAAAGCTAGGCTAGGTAC
$\bar{O}_3$	CGATAAGCTCGAATTTCGAT

Table 3: Encoding a graph in DNA [1]

5.2 Standardized Components

From a system point of view, BioBricks can be considered as active elements generating signals (proteins) when stimulated by a control signal (a protein of a certain shape). Structurally, a BioBrick is a specialized DNA strand. The mechanism according to which the BioBrick converts inputs to outputs is transcription and translation [11].

As stated earlier, a Von Neumann bio-computer requires four parts: the input and output device (IO), arithmetic logic unit, control unit and memory. The IO component serves as a means of transmitting input to other parts of the system and to the output. The logic unit performs logical operations on bus (transmissions between units) which connect these units. The IO, logic and control units work in tandem to make up the central processing unit. Program instructions are decoded by the control unit, and transformed into control signals which activate other system parts, changing the system state. Each of these units are made up of many electrical circuits, which are turned on (1) or off (0) by switches. Operations on these logic inputs are performed by logic gates to produce a logic output. This makes switches and logic gates the basic atomic units of the ALU and the control unit.

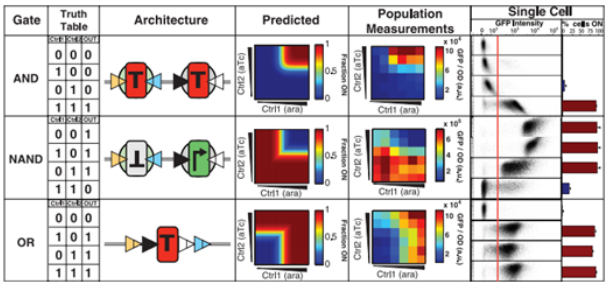


Figure 4: Defining a Signal Threshold [2]

5.2.1 *Transcriptor-Based Logic Gates*

The key component behind evaluation of logic operations in biological systems is the transcriptor - a biological transistor. Since DNA memory storage has been demonstrated, and the transmission of information throughout a biological system has been accomplished with signaling molecules such as regulatory proteins [13], the transcriptor is the final part necessary for a bio-computer. Transcriptors are used to create logic gates. These transcriptor-based gates are called "Boolean Integrase Logic" or BIL gates. BIL gates have the advantage of replicating many traditional gates in a single-layer fashion - that is, without requiring multiple instances of the simpler gates to build up more complex ones. This results in a more condensed architecture than the gates of classical computer architecture. Transcriptors are built with a careful combination of enzymes that control the flow of RNA polymerase along strands of DNA. The chosen enzymes function in bacteria, fungi, plants and animals so that bio-computers can be engineered within a variety of organisms. To draw analogy to the silicon equivalent, DNA could be seen as the wire and RNA polymerase the electron. The transcriptor performs signal amplification, which allows signals to be relayed among larger groups of cells. When activated by a control signal (in the form of specialized enzymes), they can also halt the progress of an RNA polymerase signal. The transcriptor uses a transcription terminator to block the progress of the RNA polymerase in one direction. Groups of transcriptors can accomplish computing of nearly any sort, including counting and comparison.

5.2.2 *Signal Threshold*

It is important to note, however, that the transcriptor's output is not a perfectly distinct on/off, but rather low/high cell activity. The levels of cell activity for a given logic gate are not always the same, either. However, there is a threshold over which low and high activity can be reasonably segregated, as shown in figure 4. The figure also describes the expected and empirical cell activity, reasonably in line with each given truth table. The design of BIL gates was released as public domain by its Stanford inventors to speed adoption and advancement. Computing via transcriptor is still very slow, taking a few hours between receiving an input signal and generating an output.

5.3 Limitations

In microelectronics, signals can be physically separated, but many biological devices so far lack this separation, limiting the building of reusable modules. Wiring several logic gates together can be difficult, as connections need to be implemented by a different molecule. One potential solution to this problem is by use of multiple cells following the distributed computing paradigm of silicon computers. The distributed system consists of multiple cells that communicate through a network. This paradigm is more amenable to scaling up. In the meantime, hybrids of electronic semiconductors and biological machines are being explored [13]. While bio-computers have no new capabilities from the standpoint of computability theory (for problems whose required space grows exponentially with the problem's size, space grows exponentially on both silicon and DNA computers) their potential is quite vast.

6. CONCLUSIONS

The majority of the proof-of-concept work for synthetic biology has now been accomplished. Digital DNA storage has been proven to work for bitstreams of arbitrary size, and is currently a powerful medium for large-scale archiving. DNA has been applied successfully in the implementation of the OTP, addressing it's primary drawback. Finally, the necessary components of a working bio-computer in the form of a finite state automata have been built. Now that the building blocks of a biological computer are in place, the next great challenge is the organization of these parts into an integrated system. Such a system could have much more computing power than silicon equivalents, simply by virtue of packing far more transcriptors into a smaller space. A crucial advantage which bio-computers hold over their silicon counterparts is the potential to self-organize and self-replicate, reducing engineering overhead.

7. ACKNOWLEDGMENTS

Sincere thanks to Peter Dolan, Elena Machkasova, Chris Thomas and Chase Ottomoeller for their indispensable feedback and unwavering support.

8. REFERENCES

[1] L. M. Adleman et al. Molecular computation of solutions to combinatorial problems. *Science-AAAS-Weekly Paper Edition*, 266(5187):1021–1023, 1994.

[2] J. Bonnet, P. Yin, M. E. Ortiz, P. Subsoontorn, and D. Endy. Amplifying genetic logic gates. *Science*, 340(6132):599–603, 2013.

[3] Y. Chang, C. Chen, H. Chen, I. Lin, W. Luo, C. Yang, Y. Lin, and C. Chang. Bioinformatics analysis for genome design and synthetic biology. In *Emerging Information Technology Conference, 2005.*, pages 2 pp.–, Aug 2005.

[4] J. Chen. A dna-based, biomolecular cryptography design. In *Circuits and Systems, 2003. ISCAS '03. Proceedings of the 2003 International Symposium on*, volume 3, pages III–822–III–825 vol.3, May 2003.

[5] G. M. Church, Y. Gao, and S. Kosuri. Next-generation digital information storage in dna. *Science*, 337(6102):1628, 2012.

[6] N. Goldman, P. Bertone, S. Chen, C. Dessimoz, E. M. LeProust, B. Sipos, and E. Birney. Toward practical, high-capacity, low-maintenance information storage in synthesized dna. *Nature*, 2013.

[7] D. Heider and A. Barnekow. Dna-based watermarks using the dna-crypt algorithm. *BMC Bioinformatics*, 8(1):176, 2007.

[8] Illumina, 2014.

[9] S. Jiao and R. Goutte. Code for encryption hiding data into genomic dna of living organisms. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on*, pages 2166–2169, Oct 2008.

[10] M. Madec, Y. Gendrault, C. Lallement, and J. Haiech. A game-of-life like simulator for design-oriented modeling of biobricks in synthetic biology. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 5462–5465, Aug 2012.

[11] M. Madec, C. Lallement, K. Karstens, S. Dittman, M. Gersbacher, R. Sorg, M. Wild, M. Muller, P. Bourgine, M. Donzeau, and J. Haiech. Synthetic biology and microelectronics: A similar design flow. In *Circuits and Systems and TAISA Conference, 2009. NEWCAS-TAISA '09. Joint IEEE North-East Workshop on*, pages 1–4, June 2009.

[12] D. A. Malyshev, K. Dhami, T. Lavergne, T. Chen, N. Dai, J. Foster, C. I. R., and F. E. Romesberg. A semi-synthetic organism with an expanded genetic alphabet. 509(7500):388, 2014.

[13] G. Moe-Behrens. The biological microprocessor, or how to build a computer with biological parts. *Computational and structural biotechnology journal*, 7, 2013.

[14] A. Nayebi. Fast matrix multiplication techniques based on the Adleman-Lipton model. *ArXiv e-prints*, Dec. 2009.

[15] Seagate. Seagate ships world's first 8tb hard drives. *Seagate*, Aug 2014.

[16] J.-J. Shu, Q.-W. Wang, and K.-Y. Yong. Dna-based computing of strategic assignment problems. *Phys. Rev. Lett.*, 106:188702, May 2011.

[17] R. Shukla, H. Prakash, R. Bhushan, S. Venkataraman, and G. Varadan. Sampurna suraksha: Unconditionally secure and authenticated one time pad cryptosystem. In *Machine Intelligence and Research Advancement (ICMIRA), 2013 International Conference on*, pages 174–178, Dec 2013.

[18] R. Weiss. Synthetic biology: from bacteria to stem cells. In *Design Automation Conference, 2007. DAC '07. 44th ACM/IEEE*, pages 634–635, June 2007.

[19] X. wen Chen. Computational models in systems biology. In *Granular Computing, 2009, GRC '09. IEEE International Conference on*, pages 1–1, Aug 2009.