An Overview of Techniques and Applications of DNA Nanotechnology

Technical Report UT-CS-11-682

Kristy S. Van Hornweder*

Department of Electrical Engineering & Computer Science
University of Tennessee, Knoxville
kvanhorn@utk.edu

September 27, 2011

Abstract

In this report, a number of mechanisms and computational applications using DNA nanotechnology are presented. A brief biological background is given to introduce several operations on DNA. Following the background information is a discussion of three DNA-based technologies: DNA computing, DNA tilings, and DNA machines. The advantages of DNA computing are discussed, as well as the use of DNA computing in solving a classic graph routing problem. Examples of computation using DNA tilings are presented, including a binary counter and XOR cellular automata. DNA machines such as tweezers, gears and wheels, and walkers are also discussed and illustrated. A brief discussion of issues and challenges for all three types of DNA nanotechnology also appears. In addition, several comparisons of DNA-based technology to conventional silicon-based technology are given.

1 Introduction

DNA nanotechnology is a current area of research where computing and technology is applied at the molecular level, as opposed to the level of conventional electronic computers. The field of DNA-based computers was born in 1994, when Leonard Adleman used DNA to solve a simple version of a classic graph routing problem. Conventional silicon chips may likely reach a limit in terms of speed and miniaturization. In order to achieve even faster speeds and smaller sizes, a new technology will need to be developed. A biochemical solution of DNA molecules and enzymes could be embedded into future computer chips. DNA-based computing technologies have potential use in a wide variety of applications, including computation, nanofabrication, nanoelectronics, nanorobotics, and nanomechanical devices. This paper gives a brief overview of several approaches to DNA-based nanotechnology: DNA computing, self-assembly of DNA tilings, and DNA-based machines.

2 Biological Background

While reading James Watson's book "Molecular Biology of the Gene" [15], Adleman realized that computers and living cells store information in a similar way. Computers store data as strings of 0s and 1s, whereas cells store information for the development and functioning of an organism as chains of the code letters A, T, C, and G. Each of the four base letters represents a nucleotide, which is composed of a nitrogenous base, a sugar, and a phosphate group. Nucleotides are chained together to form strands of DNA. Two single strands of DNA can be formed into a double strand by matching complementary base pairs. The bases A and T are complements, as are C and G. Thus, an A nucleotide on one strand will bond with a T nucleotide on the other strand, and similar for C and G. An example illustrating DNA base pairings is given in Figure 1.

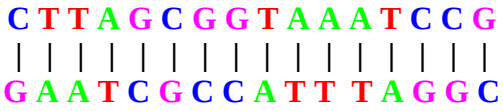

Figure 1. DNA base pairings.

DNA strands are manipulated by various biocatalysts called enzymes. DNA strands are joined together through a process known as ligation, which is catalyzed by a ligase enzyme. Ligation is often used for repair and replication. The strands are joined by hybridizing complementary base pairs. Figure 2 illustrates the ligation process.

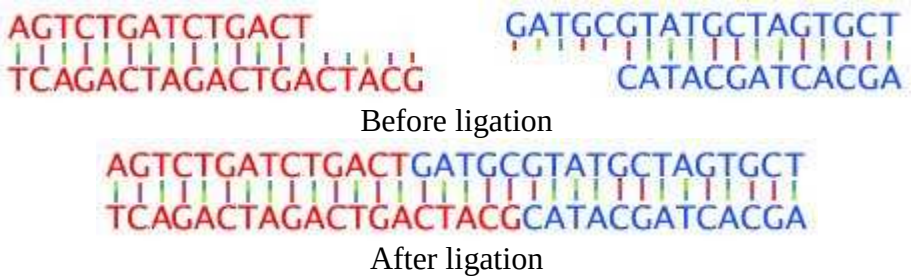

Figure 2. DNA ligation [22].

Polymerase enzymes are used to copy a DNA strand or a segment of a DNA strand for amplification. In the process of cleavage, a DNA strand is cut by using a restriction enzyme. These enzymes do not cut a DNA strand at any place, rather, they must bind with a certain sequence of base pairs. The strands are separated by disconnecting the chemical bonds of complementary base pairs. Cutting a strand of DNA often results in a sticky end, which is a series of unpaired nucleotides at the end of a DNA strand. Sticky ends control how DNA strands are joined together. Figure 3 shows how a DNA strand is cut by a restriction enzyme.

Figure 3. Restriction (cleavage) of DNA [22].

A series of operations such as joining, copying, and cutting are performed on DNA strands by enzymes to carry out the process of computation or mechanical function. The processes of ligation and cleavage are used in applications such as DNA walking devices, in which the feet of a walking structure bind and unbind to a track that the device walks along. The operation of the DNA walker models the functioning of the protein kinesin. Kinesin is a motor protein that transports substances in a cell (such as chromosomes or cell organelles) by walking unidirectionally along a microtubule track, as shown in Figure 4. The walking process is driven by the hydrolysis of ATP, which is a molecule that supplies energy to be used in metabolism.

Figure 4. Kinesin attached to a microtubule track [22].

The idea is to design molecular processes such as DNA computation so that they are truly autonomous. It is desired to simply create a chemical solution consisting of the necessary DNA strands and enzymes and then allow the reactions to occur sequentially on their own, without any need for external intervention.

More information about biochemical concepts related to DNA nanotechnology can be found in the Appendix.

3 DNA Computing

In 1994, Leonard Adleman [1] successfully implemented the first molecular computation, in which he solved a seven city Traveling Salesman Problem using DNA. The experiment was inspired by James Watson's book "Molecular Biology of the Gene" [15]. Three years later, DNA logic gates were developed. These logic gates detect certain DNA strands as input, and join them together to form a single output. DNA molecules offer compact storage and miniaturization well beyond that of conventional computers, and DNA computing is able to exploit a massive level of parallelism unattainable by current computer technology.

3.1 Advantages of DNA-Based Computers over Silicon-Based Computers

Massive parallelism: Instead of working on one DNA molecule at a time, enzymes in the solution can work on numerous DNA molecules in parallel, which could theoretically result in 10 trillion calculations performed simultaneously. Even though the asymptotic complexity of an algorithm would not be reduced, because of the potential parallelism, DNA computing techniques may be able to solve complex mathematical problems within hours, whereas it would likely take hundreds of years for current electrical computers to solve the same problem. While the fastest current supercomputers can process $128*10^{15}$ bits in 1000 seconds, in theory, a test tube of DNA molecules could process 10^{18} bits in 1000 seconds [5].

Compact storage: Because of the molecular size of DNA, DNA computers would be able to store billions of times more data than conventional computers. One pound of DNA can store more information than all of the electronic computers ever built [3]. The data density of DNA is over 1 million Gbits per square inch, whereas a high performance hard drive can hold 7 Gbits per square inch [13]. To put this in perspective, 1 gram of DNA the size of a half-inch sugar cube could store as much information as a trillion CDs [21].

Miniaturization: DNA computers can be many times smaller than conventional computers. More than 10 trillion DNA molecules can fit into an area no more than one cubic centimeter [3].

Less energy required: Adleman's experiment achieved a rate of $2*10^{19}$ operations per joule. The theoretical limit as determined by the second law of thermodynamics is $34*10^{19}$ operations per joule. Modern supercomputers are able to achieve a rate of 10^9 operations per joule [12].

Manufacture of biochips is cleaner: Toxic substances are used to produce conventional silicon chips [3].

Constant supply of DNA: Provided that there are living cellular organisms, there will always be a supply of DNA available [3].

Cheaper resource: Since there is a large supply of DNA, raw materials for producing biochips are cheaper [3].

3.2 Biological Operations

Various operations can be performed on DNA strands to implement computational algorithms, as given below [5, 6].

Extract: Extract strands of DNA containing a specific nucleotide sequence and separate them from the rest of the strands
Merge: Combine two tubes of solution without changing any individual DNA strands
Discard: Discard the tube
Amplify: Make copies of DNA strands or sections of strands
Anneal: Let single DNA strands join to form double strands
Cut: Use a restriction enzyme to cut a DNA strand
Join: Join DNA strands together
Append: Elongate a DNA strand by adding a short strand onto its end
Append head: Same as Append, except add the short strand to the beginning instead of the end
Length: Separate the DNA strands by length
Detect: Determine if a tube contains a DNA molecule or not
Read: Give a description of a specific molecule

3.3 Adleman's Traveling Salesman Experiment

3.3.1 Problem

Given a set of cities and paths between cities, find a route that begins in the starting city, ends in the destination city, and visits all the cities exactly once. Figure 5 gives a pictorial view of the problem.

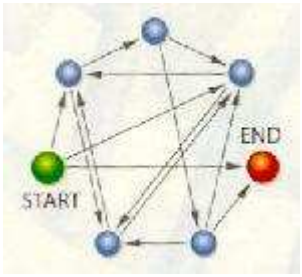

Figure 5. Graph of paths between cities [9].

3.3.2 Algorithm [1, 13]

1. Generate random paths through the graph

Each vertex (city) and edge (path) is encoded using a DNA strand that is 20 nucleotides long. The strands for the vertices are chosen randomly. An edge strand is chosen so that the first 10 nucleotides are the complement of the last 10 nucleotides of vertex 1, and the last 10 nucleotides

are the complement of the first 10 nucleotides of vertex 2. This is shown in the example in Figure 6. For simplicity, assume the two cities are encoded with 10 nucleotides rather than 20. Suppose Knoxville TN is encoded as ACCTTGAAGC and Asheville NC is encoded as GTAGCCCATA. Then the edge between Knoxville and Asheville will be encoded as CTTCGCATCG.

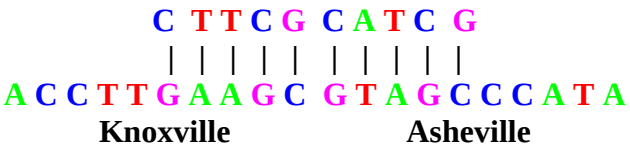

Figure 6. DNA encoding of vertices and edges.

The length of the strands must be long enough so that each will uniquely encode a vertex and so that no two vertices share a long subsequence.

2. Only keep paths that begin with the start vertex and end with the destination vertex

In this step, the DNA sections that start and end with the appropriate vertices are copied and amplified using a process called polymerase chain reaction (PCR). Primers that are complementary to the start and end vertices are used to mark each end of the strand section to be copied.

3. Only keep paths with exactly *n* vertices, where *n* is the number of vertices in the graph

The idea of this step is to sort the DNA strands by their length, and select the strands that correspond to the correct number of vertices. A technique called gel electrophoresis is used, where DNA strands are forced through a gel at different speeds, depending on the length of the strands. The result is a series of DNA bands, and the band corresponding to the length of interest can be isolated. Figure 7 shows the result of gel electrophoresis, where the lengths of the DNA strands are in terms of the number of base pairs.

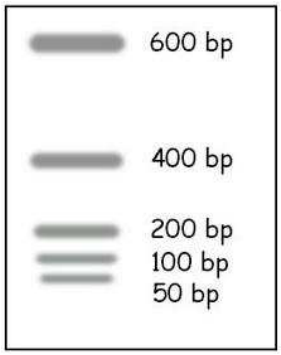

Figure 7. Lengths of DNA strands as given by gel electrophoresis [13].

4. Only keep paths that go through each vertex at least once

The strategy is to look up each vertex in a DNA strand. For each vertex, a bead corresponding to the complement of that vertex will attach to the DNA strand if the strand contains that vertex. Each time a vertex is found in the strand, the strand is filtered out. The strands that remain are the strands containing all of the vertices. Figure 8 shows an example of applying this technique. A bead corresponding to the complement of New York binds with the DNA strand representing New York, indicating that New York is in the path of cities.

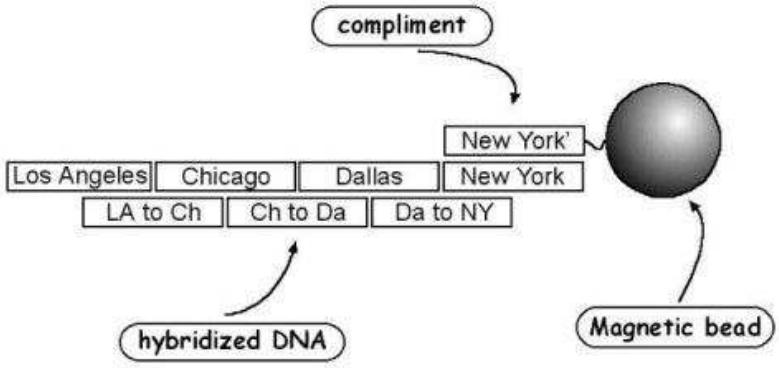

Figure 8. Finding a vertex in a path [13].

5. If any DNA strands remain, then there is a solution path

To readout the sequence of vertices in the path, a graduated PCR method is used. Primers for the starting vertex and for each of the other vertices in turn are attached to the proper locations on the DNA strand. In this way, the length of the DNA section can be measured, and this length indicates the position of the vertex in the path sequence.

3.3.3 Analysis

Although Adleman's technique successfully computed the solution to the Traveling Salesman Problem, it required seven days of lab work to obtain the solution. Human intervention was required between each step. All of the possible routes through the graph could be generated within seconds, but days were required to narrow down the possibilities. In theory, using automated techniques would significantly reduce the time required to solve the problem.

According to [6], an algorithm is rated by the number of biological steps required, and the number of DNA strands used. The length of the strands is typically linear in the problem size, while the number of strands is exponential. In terms of practicality, 10^{21} (2^{70}) is an upper bound on the number of DNA strands that an algorithm can use.

Another issue concerning algorithms using DNA is the error rate. DNA molecules are fragile, and enzymes occasionally make mistakes in copying or cutting strands. It is possible for correct paths to not bind together and also for incorrect paths to bind together, thus forming pseudopaths. Possible remedies are to implement stricter amplification or separation techniques.

3.4 Applications

Graph and routing algorithms. In addition to the Traveling Salesman Problem, methods have been proposed for solving the clique problem, graph coloring problems, the independent set problem, and other graph and set problems.

Boolean Satisfiability. The general problem states: given a logical expression, find a truth assignment for the variables to satisfy the expression. A 4-SAT version of the problem has been solved [7] by encoding all possible assignments on a 2D surface and using restriction enzymes to remove assignments that do not satisfy the expression. Adleman and colleagues were able to solve a 20-SAT problem [4] using automated gel electrophoresis to separate the strands that satisfy the expression. One practical application of these methods may be for executing Boolean retrieval queries into a genome database [10]. The extent of using DNA to solve satisfiability problems is limited by the number of DNA strands required. A possible upper limit may be 70-80 Boolean variables [10].

Cryptography. The security of the RSA public-key cryptosystem depends on the difficulty of factoring a product of two large prime numbers. Methods are proposed in [5] for a DNA-based parallel subtractor, parallel comparator, and parallel modular arithmetic.

Medical applications. Eventually, DNA computing algorithms could be used in numerous medical applications such as diagnosis and treatment of disease, maintaining the health of astronauts on deep-space voyages, and providing a better understanding of the organization and functioning of the human brain.

3.5 Issues and Challenges

There are several challenges that face researchers in the field of DNA computing that must be considered before algorithms using DNA molecules can become practical.

Error rate: Computations using DNA are prone to errors due to the fragile nature of DNA molecules, enzymes making errors in copying, cutting, and inserting nucleotides, and damage from radiation. By using the fact that each nucleotide of a DNA strand has a complement nucleotide, one possible scheme for handling errors could be to use a type of redundancy similar to RAID 1 technology, in which redundant disks are used for recovering lost data.

Automation: One major drawback is that an individual biological operation performs quite slowly. Implementing the biological steps so that they are automated may have a significant impact on the speed of the computation process. The operations would also be less labor intensive since there would be no need for human intervention during and between steps. Automated techniques could possibly employ self-assembly of DNA strands, or involve a solution containing the required restriction enzymes and ligases that could execute their functions on their own, without any need for external stimulation.

Number of DNA strands required: Oftentimes, the number of DNA strands in a computation grows exponentially with the problem size. Beyond a certain point, the computation may require too many strands to be considered practical.

DNA molecules used in a computation are specialized: Different types of problems will require different encodings of input into the DNA strands. Thus, the same DNA strands cannot be reused in later computations.

What other biological operations are possible? There may possibly be other operations not described above that could operate on DNA molecules. Perhaps different types of operations or combinations of existing operations could lead to more efficient algorithms in terms of speed and the amount of DNA required.

4 DNA Tilings and Self-Assembly

Self-assembly of DNA molecules is potentially applicable in many areas of nanotechnology, including fabrication of nanostructures, nanoelectronics, nanorobotics, and autonomous nanomachines. DNA self-assembly could also be used to perform computation since the tiling process theoretically simulates operations of a Turing machine. Combinatorial search problems such as the Hamiltonian Path Problem and the SAT problem could be solved in this manner, however, error rates and low speed of computation indicate that DNA self-assembly shows more promise in material science than in computation. In particular, self-assembly of DNA may prove to be the most useful in nanoelectronics, especially since DNA has the advantages of miniaturization and precision over current silicon-based technologies.

4.1 DNA Tile Structures

A tiling is an arrangement of shapes that fit together perfectly in the infinite plane, similar to the way the pieces of a jigsaw puzzle interlock. Numerous configurations of DNA tiles have been constructed, including crosses, hairpins, wire-frame cubes, octahedrons, figure 8s, rings, triangles, and double crossover structures (DX) [17]. These tiles can assemble into a 2D grid-like lattice, which can then be used as a nanocircuit or to perform computation. Figure 9 shows the DNA cross and DX structures.

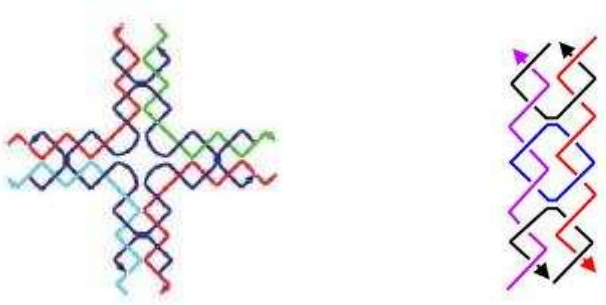

Figure 9. DNA cross (left) [18] and DX (right) [8] structures.

Structures such as nanoribbons and 2D nanogrids can be used as a framework for the organization and layout of nanowires and electronics. A barcode lattice representing a bit pattern uses DX DNA tiles bonded to a scaffold DNA strand. A tile with a stem loop represents a 1 while a 0 is denoted by a tile without a stem loop [19]. A schematic drawing of the barcode lattice appears in Figure 10.

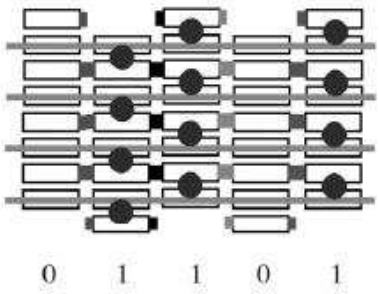

Figure 10. DNA barcode lattice [19].

4.2 Computing with Tiles

To illustrate the structure of a lattice formed by arbitrary tiles, consider Figure 11. The lattice shown here consists of two types of tiles, A and B. The shape of each type of tile determines how the tiles are pieced together and thus the pattern of the lattice structure. Each tile consists of four binding regions: upper left, lower left, upper right, and lower right. Changing the configuration of these regions will change the way the tiles fit together. The lattice

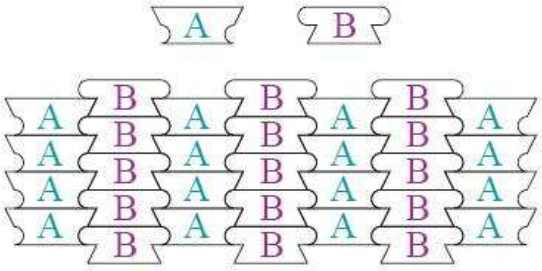

Figure 11. A lattice formed from two types of tiles [16].

pattern and tile configurations define the type of computation that is performed. The four binding regions of a tile consist of sticky end sequences of a double crossover DNA strand. An example of binding regions with sticky ends is depicted in Figure 12. The two DNA molecules shown here correspond to the A and B tiles of Figure 11. For instance, the upper right region of the A molecule (CATAC) is the complement of the lower left region of the B molecule (GTATG), hence, these regions of the two molecules will bind.

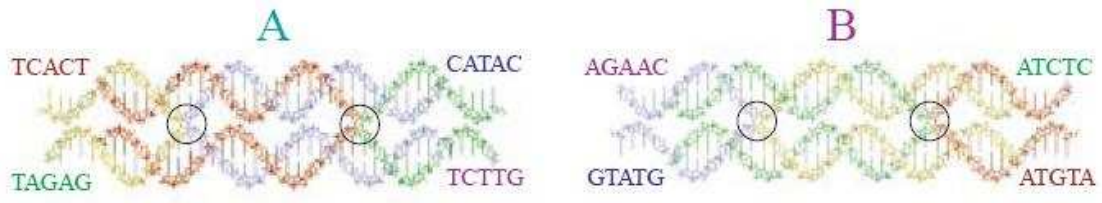

Figure 12. Sticky end configuration of A and B tiles [16].

Another example of computing with DNA tiles is given in Figure 13. The bottom row of tiles represents the set of input values into the system. The program tiles at the very top of the figure resemble operations or rules of the computation. The tiles are then assembled according to the appropriate bindings, and the tiling process eventually results in the output sequence, which is shown as the top row of the lattice. It is possible that there may be several different valid tilings. These multiple possibilities could generate a combinatorial library of tilings [16], each of which can be worked on simultaneously, thus exploiting the parallelism available to DNA computing.

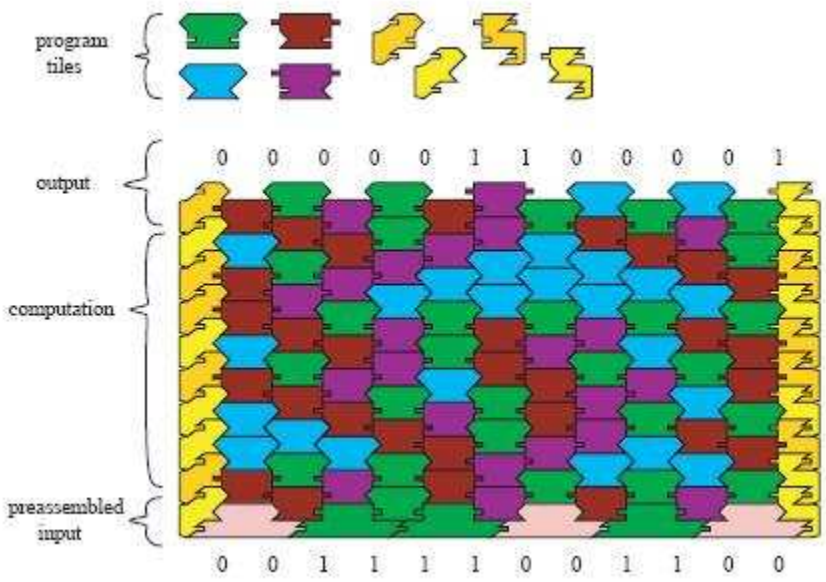

Figure 13. Structure of lattice used for computation [16].

4.3 Binary Counter Example

A binary counter implementation using DNA tiles is shown in Figure 14. The bottom row represents the first number, and each successive row in the upward direction encodes the binary number representing the next natural number. The lattice begins with a seed tile in the lower right corner, and continues with anchor tiles on top and to the left of the seed tile. The lattice grows upward and to the left to carry out the counting process. The top and bottom portions of each tile determine whether that tile represents a 0 or 1, and the left and right sides of a tile handle rollover from the previous bit. Rollover corresponds to a change in place value for decimal numbers, in which a 9 changes to a 0 and the digit in the next place to the left is increased by one. If the right edge of a tile indicates rollover, then the tile will contain the opposite value of the bit in the tile below. If there is no rollover from the right edge, then the tile

keeps the same bit value as the tile below. Modifying the pattern of the edges of the rule tiles will allow different mathematical operations to be performed. In this way, the tiling process is Turing universal.

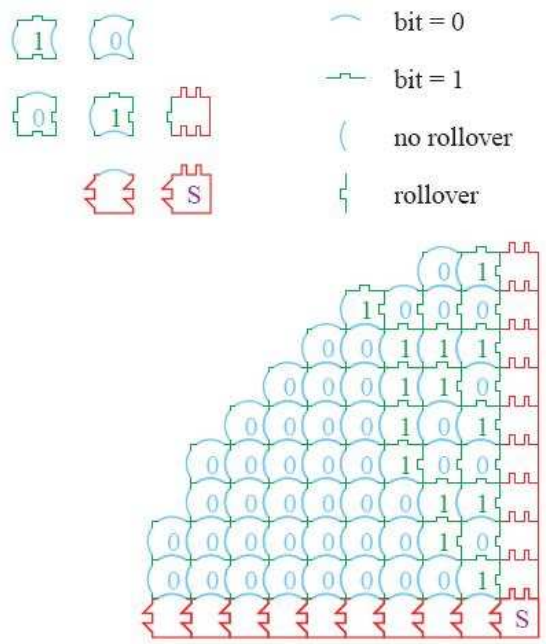

Figure 14. Binary counter implementation using tiles [16].

4.4 XOR Cellular Automaton Example

A tiling scheme that implements an XOR cellular automaton is depicted in Figure 15. The arrangement of the tiles constructs the Sierpinski triangle fractal pattern, which is shown in the upper right corner of the figure. In Section A, three time steps are shown, with time beginning on the bottom row. Odd and even rows are interleaved and the arrows represent the propagation of information from one step to the next. Section B depicts the structure of the individual tiles and illustrates how they are pieced together. The bottom corners of the tile represent the inputs x and y, while the output z is represented by the upper corners of the tile. The XOR equation is given by $z=x \oplus y$. Section C of the figure shows the four different rule tiles that represent the four combinations of input values for x and y along with their output value: $0 \oplus 0=0$, $1 \oplus 1=0$, $0 \oplus 1=1$, and $1 \oplus 0=1$. Rounded portions of the tiles denote a 0, whereas rectangular notched regions represent a 1. Tiles with an output value of 0 are shown in gray, while tiles having an output value of 1 are shown in white. The bottommost row of the lattice structure (in blue) represents the initial conditions for the computation. The red asterisks below this initial row denote a binding region of 1, as indicated by the white tile connected to this position in the bottom initial row. All other sites on the initial row indicate binding regions of 0. Section D shows a tiling that is free of errors, and in Section E, the displayed tiling contains four tile mismatch errors, as indicated by the four tiles marked with an X.

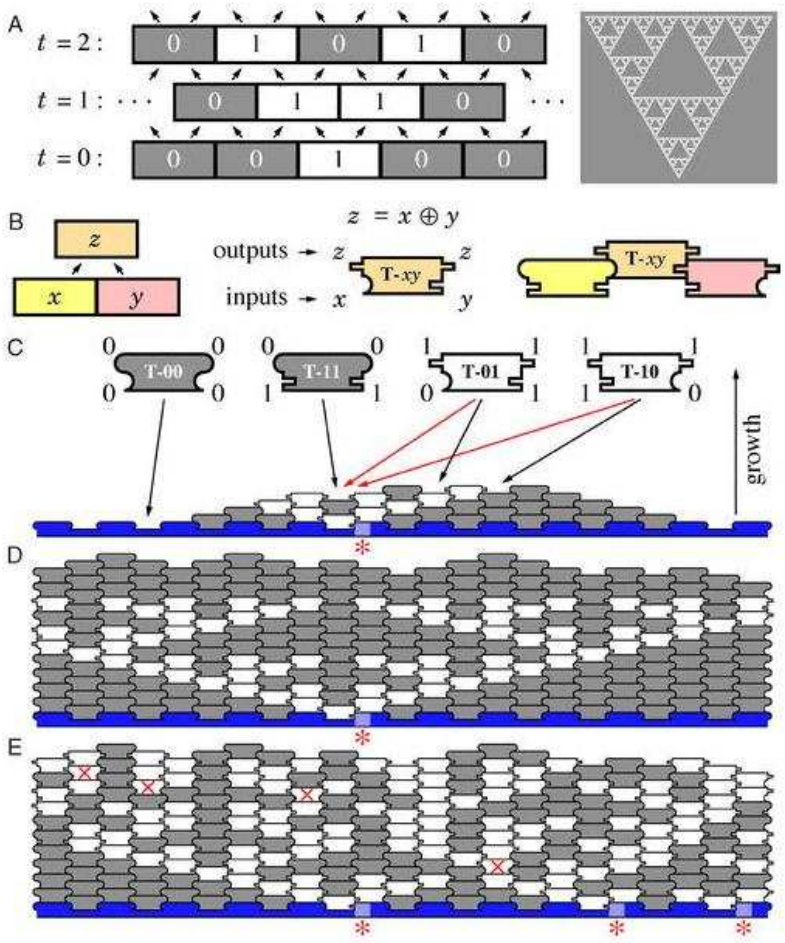

Figure 15. XOR cellular automaton implementation using DNA tiles [11].

4.5 Issues and Questions

There are several issues that must be considered before self-assembly using DNA tilings can be applied in practice. One major concern is that error rates from 1 to 10% per step have been observed [17]. Due to defects in tiles, the lattices become slightly distorted, thus contributing to higher error rates. One possible method of reducing errors that has been proposed is the design and application of error-correcting tiles. Another issue is the undesired nucleation that often results from the tiling process. Nucleation occurs when molecules bind anywhere they match, even if the bond is relatively weak, or a region only partially matches. The result is that tiles are not assembled properly, thus causing incorrect execution of the computation process. A third issue concerns the fact that linear self-assembly is decidable, meaning that only simple computations can be performed. More interesting computations require the use of two or three dimensional lattices, since tiling problems of these dimensions have been proven to be undecidable, and thus able to simulate computation using a Turing Machine.

An open problem in the field is the design and creation of 3D arrays of DNA tiles. The implementation of these structures would allow for more interesting and efficient computation.

Other open questions include: what kinds of shapes and patterns can be assembled using DNA tiles, and how quickly can these patterns be assembled [17].

5 DNA Machines

The properties of DNA allow for the creation of molecular devices that perform machine-like functions and act as walkers, sensors, and motors. Introducing chemical substances such as enzymes will trigger the DNA strands to rearrange their chemical bonds and undergo structural changes, which will allow the devices to function as machines. Such machines have many applications in nanorobotics and the fabrication of nanostructures.

5.1 DNA Tweezers

DNA tweezers could be used in artificial limbs of nanorobots for grasping or holding objects. One possible configuration of tweezers contains two double-stranded DNA structures joined by a single-stranded piece of DNA, as shown in Figure 16. A biocatalyst binds with the single strand, which forces the tweezers structure to open. The single strand is then cut by the catalyst and the tweezers returns to its closed state. In the presence of substrate DNA strands and a biocatalyst, the tweezers can alternate between its open and closed states.

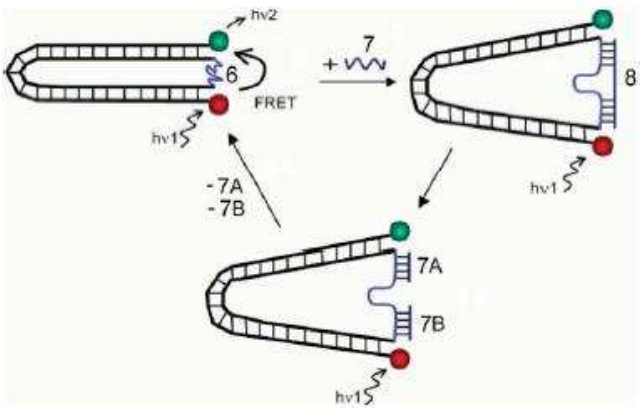

Figure 16. DNA tweezers [2].

5.2 DNA Gears and Wheels

DNA gears have potential use in nanostructures such as pumps or wheeled vehicles. A gear consists of a circular DNA structure and motion is achieved by two of these structures rotating against one another. DNA-based gears are shown in Figure 17. Each structure contains sticky ends around the wheel that bind with complementary sticky ends of another wheel. The sequential connection and disconnection of these sticky ends results in rotational motion.

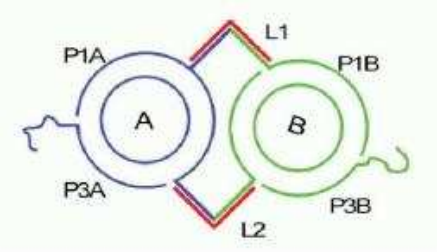

Figure 17. DNA gears [2].

5.3 DNA Walkers

Perhaps the most intriguing nanomachine is a DNA walking device that could be used for transporting nanosubstances or performing computation. The operation of the walker is modeled after the functioning of proteins such as myosin and kinesin. Myosin is responsible for contracting muscles, and kinesin transports and places proteins and organelles within a cell.

The DNA walker moves along a nucleic acid track one step at a time. The track consists of double-stranded DNA and contains single-stranded footholds protruding from the surface. A schematic diagram of a DNA walker appears in Figure 18. The walker attaches to the end of foothold A by attaching to complementary sticky ends. A ligase enzyme then causes footholds A and B to hybridize, forming the A-B complex. By binding to a recognition site, a restriction enzyme cuts this complex, which results in the walker being attached to foothold B. Another process of ligation joins footholds B and C into a B-C complex. A second restriction enzyme cleaves the B-C complex, which results in the walker attaching to foothold C, thus completing one cycle of the walking process.

The ligation and cleavage processes occur in such a way that backward motion of the walker is prevented, as explained in [20]. When the walker is cut from a foothold by an endonuclease, the sticky ends of both the walker and the foothold are changed. The sticky end of the walker is complementary to that of the foothold immediately ahead of the walker, but not complementary to that of the foothold directly behind the walker. Thus, the walker can only be ligated with the foothold directly in front of it, and not with the one directly behind it. This arrangement of complementary sticky ends guarantees the unidirectional motion of the walker.

The walker is also autonomous, in that once the needed enzymes are present, the walking process occurs spontaneously, without any need for external intervention. The ligation and cleavage process described above will occur even when all enzymes are simultaneously present. One goal in applying this process is to devise a machine that takes as input a set of instructions for designing a device, and produces as output a listing of DNA sequences that will assemble into the desired device [14].

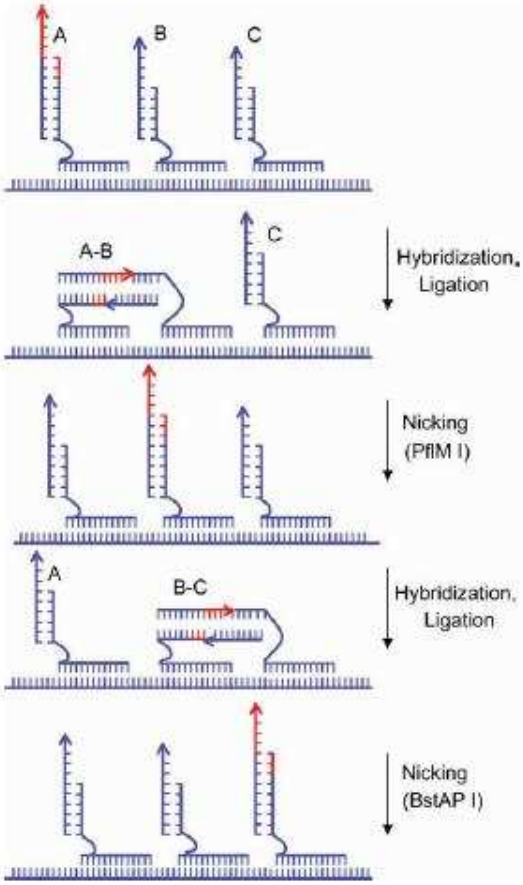

Figure 18. DNA walker [2].

There are several functions of DNA walking devices that are still scientific goals. A few of them are discussed briefly below [2].

1. **Allowing for multiple rounds of walking**. In the example given above, the walker started on foothold A, moved to foothold B, and subsequently to foothold C. Enabling the walker to move an indefinite amount of steps is a necessary goal in order for DNA walkers to be practical.

2. **Transferring a walker to another track**. This ability will allow a walking device to move along a different path that branches from the path it had previously been walking along. In this way, the walker will be able to navigate through a series of different pathways.

3. **Designing walkers with the ability to choose a path or direction based on external triggers**. The walker would be able to make a decision depending on factors such as light, heat, chemical composition of the environment, or the presence of a substance, such as a toxin.

4. **Multiple walkers following in one another's footsteps**. Functionality of this type would be employed in nano assembly lines, which consist of a sequence of walkers each transporting one nanoparticle at a time.

6 Conclusions

While the field of DNA nanotechnology in still in its infancy, recent breakthroughs have shown that DNA-based technology is promising, and it is possible that DNA computers could begin to replace silicon-based computers in the next decade. With DNA computers, unprecedented levels of massive parallelism, compact storage, miniaturization, and speed of computation could be achieved. The development of areas such as nanofabrication and DNA computing has the potential to take applications such as space exploration and the use of medical devices to a whole new level of sophistication.

References

[1] Adleman, L. Molecular Computation of Solutions To Combinatorial Problems, *Science*, Vol. 266, pp. 1021-1024, 1994.

[2] Beissenhirtz, M., Willner, I. DNA-based Machines, *Organic and Biomolecular Chemistry*, Vol. 4, pp. 3392-3401, 2006.

[3] Bosner, K. How DNA Computers Will Work. http://electronics.howstuffworks.com/dna-computer.htm

[4] Braich, R.S. et al. Solution of a 20-Variable 3-SAT Problem on a DNA Computer, *Science*, Vol. 296, pp. 499-502, 2002.

[5] Chang, W., Guo, M., Ho, M. Fast Parallel Molecular Algorithms for DNA-Based Computation: Factoring Integers, *IEEE Transactions on Nanobioscience*, Vol. 4, No. 2, pp. 149-162, June 2005.

[6] Lipton, R. et al. On the Computational Power of DNA, *Discrete Applied Mathematics, Special Issue on Computational Molecular Biology*, Vol. 71, pp. 79-94, 1996.

[7] Liu, Q. et al. DNA Computing on Surfaces, *Nature*, Vol. 403, pp. 175-179, 2000.

[8] Mao, C. The Emergence of Complexity: Lessons from DNA, *PLoS Biology*, Vol. 2, No. 12, 2004.

[9] Paine, M. The DNA Computer, http://users.tpg.com.au/users/aoaug/dna_comp.html, 1998.

[10] Reif, J. Successes and Challenges, *Science*, Vol. 296, pp. 478-479, April 2002.

[11] Rothemund PWK, Papadakis N, Winfree E. Algorithmic Self-Assembly of DNA Sierpinski Triangles, *PLoS Biology*, Vol. 2, No. 12, 2004.

[12] Ruben, A., Landweber, L. The Past, Present and Future of Molecular Computing, *Nature Reviews-Molecular Cell Biology,* Vol. 1, pp. 69-72, Oct. 2000.

[13] Ryu, Will. DNA Computing: A Primer. http://arstechnica.com/reviews/2q00/dna/dna-2.html

[14] Smalley, E., Patch, K. DNA Machines Take a Walk, *Technology Research News*, 2004.

[15] Watson, J. D. et al. *Molecular Biology of the Gene*, Benjamin/Cummings Publishing Co., Menlo Park, CA, Ed. 3, 1987.

[16] Winfree, E. Algorithmic Self-Assembly of DNA: Theoretical Motivations and 2D Assembly Experiments, *Journal of Biomolecular Structure & Dynamics*, Vol. 11, No. 2, 2000.

[17] Winfree, E. DNA Computing by Self-Assembly, *The Bridge: Expansion of Frontiers of Engineering*, Vol. 33, No. 4, 2003.

[18] Yan, H. et al. Combinatorial Self-Assembly of DNA Nanostructures, *Organic Biomolecular Chemistry*, 2006.

[19] Yan, H. et al. Self-Assembled DNA Structures for Nanoconstruction, *DNA-BASED MOLECULAR ELECTRONICS: International Symposium on DNA-Based Molecular Electronics, AIP Conference Proceedings*, Vol. 725, pp. 43-52, 2004.

[20] Yin, P. et al. Designs of Autonomous Unidirectional Walking DNA Devices, *DNA Computing*, 2004.

[21] DNA Basis for New Generation of Computers, CNN article, Aug. 2003. http://www.cnn.com/2003/TECH/ptech/08/18/biological.computing.ap/

[22] Wikipedia, The Free Encyclopedia.

Appendix – Biochemical Terminology Relevant to DNA Nanotechnology [22]

ATP (adenosine triphosphate): A nucleotide that transports chemical energy in living cells to be used for metabolism. It is produced as energy during photosynthesis and cellular respiration. It is also used in replication, synthesis, and transcription of DNA.

base pairs: Two nucleotides each on a complementary DNA strand that are connected by a hydrogen bond. The base pairings of DNA are A-T and C-G.

chromosome: A very long piece of DNA that contains genetic information.

cleavage: A division, separation, or breaking of chemical bonds. Also known as restriction.

crossover region: An area where a DNA strand is split and where an end piece is exchanged with that of another DNA strand. Thus, a crossover region is involved in genetic recombination.

dehybridization: Separation of two complementary strands of DNA.

DNA: A nucleic acid that contains genetic information and instructions for the development and functioning of an organism, and the construction of components of a cell.

DNAzyme: A DNA molecule that acts as an enzyme.

duplex: A double stranded molecule of DNA or RNA.

endonuclease: An enzyme that cleaves bonds of molecules.

enzyme: A protein catalyst that accelerates a chemical reaction.

G-quadruplex: A type of DNA structure consisting of four strands arranged in a square of G's and stabilized by hydrogen bonds.

hairpin: A short stem loop in a single-stranded DNA molecule. A place where two regions of the same molecule form a double helix that ends in an unpaired loop.

hybridization: Joining of two complementary strands of DNA.

hydrolysis: A chemical process where two molecular components are joined together and a molecule of water is produced.

kinesin: A motor protein that transports substances in a cell by walking unidirectionally along a microtubule track. The walking process is driven by ATP hydrolysis. The protein may be transporting chromosomes or cell organelles (such as mitochondria).

ligase: An enzyme used to catalyze the process of ligation.

ligation: The linking of DNA strands for repair or replication.

nucleic acid: A macromolecule composed of chains of nucleotides that carries genetic information. The most common examples are DNA and RNA.

nucleotides: The building blocks of nucleic acids. They are composed of three components: a base, a sugar, and a phosphate group.

oligonucleotide: Short sequences of nucleotides used for detecting complementary DNA or RNA strands.

pH: An numerical indication of how acidic or basic (alkaline) a solution is. On a scale from 0 to 14, 0 is the most acidic and 14 is the most basic. Acidic substances (those with a low pH) include battery acid and citric acid and basic substances (those with a high pH) include ammonia and bleach. Water (pH 7) is considered neutral.

polymerase: An enzyme that catalyzes the replication and transcription of DNA or RNA.

recognition site: A place where an enzyme can bind in order to catalyze a chemical reaction.

restriction endonuclease: An enzyme used for cutting double stranded DNA molecules.

RNA: A nucleic acid that is a messenger between DNA and the ribosome of the cell, which is where proteins are built from amino acids. The genetic information from the DNA molecule is passed on to the RNA molecule, which is then translated into proteins.

stem structure: A type of DNA structure where a short strand (stem) of DNA protrudes from a longer strand.

sticky ends: Unpaired nucleotides at the end of a DNA molecule. Also referred to as a single-strand overhang. Sticky ends are often created by endonucleases after they cut a DNA strand.

substrate: A molecule that an enzyme acts on.

transcription: The translation of DNA into proteins. The DNA is first transcribed into an intermediate molecule known as RNA, which then uses the genetic information encoded in DNA to form proteins.