DNA, Protein-Structure and Turing Machines

Ihab M. E.

1 Turing Machines

A Turing machine is better than a finite state machine and a push down automaton, but that's a topic for another day. A Turing machine consists of:

- A **tape** divided into cells, each containing a symbol from some finite alphabet.
- A **head** that can read and write on the tape, and move one step at a time to the right or the left of the tape.
- A **state register** that stores the state of the Turing machine.
- A finite **table** of instructions such that given a specific state and a symbol it is reading, it tells the machine to do the following in sequence:
  - Either erase or write a symbol.
  - Move the head (L or R)
  - Go to a specific state, may be the same state it was in.

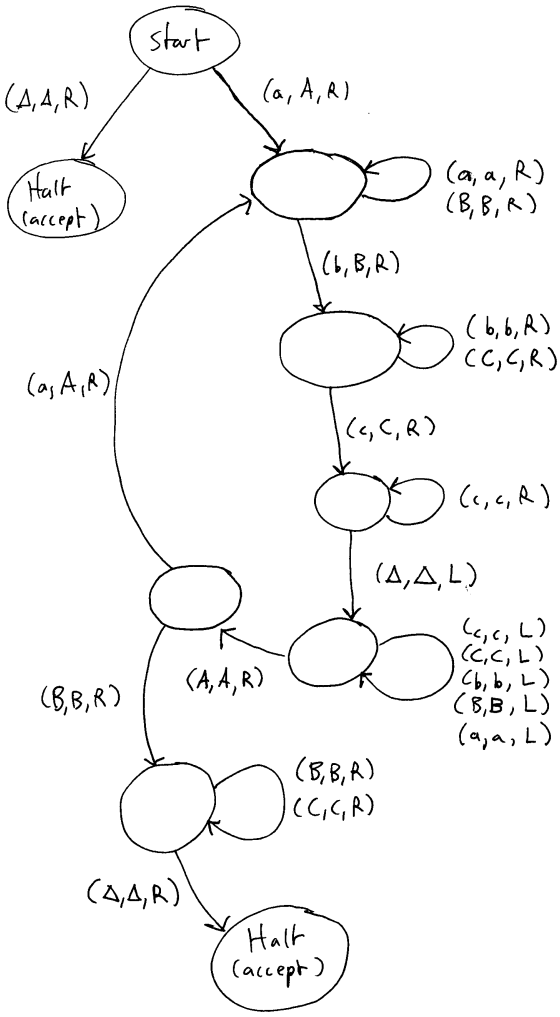
1.1 Formal Definition

A turing machine can be described by a 7-tuple $M = \langle Q, \Gamma, b, \Sigma, \delta, q_0, F \rangle$ where:

- Q is the set of states, which is finite and non-empty.
- Γ is the alphabet of the tape, which is also finite and non-empty.
- $b \in \Gamma$ is the blank symbol.
- $\Sigma \subseteq \Gamma \backslash \{b\}$ is the set of input symbols, which are allowed to appear in the initial tape contents.
- $\delta : (Q \backslash F) \times \Gamma \to Q \times \Gamma \times \{L, R\}$ is the transition function.
- $q_0 \in Q$ is the intial state.
- $F \subseteq Q$, is the set of final/accepting/halting states.

To illustrate how powerful a Turing machine is, let us examine a language which is defined by $L = \{a^n b^n c^n | n \geq 0\}$ which is neither a regular language nor a context-free one, and you can easily prove that by the pumping lemma for both context-free languages and regular languages, which means that this particular language can not be described by a finite state machine or a push down automata (**PDAs = CFGs**)

A simple turing machine to describe the language L would be the following:



1.2 Universal Turing Machines

A turing machine can be used to implement any algorithm imaginable, but that would be fine if it's just done theoretically on paper, but you would not want to implement an actual Turing machine every time you want to solve a new problem. A universal Turing machine is just a TM that is constructed specifically to take as an input the description for another Turing machine and a data tape, and it simulates that turing machine on its own tape. The computers we use today are a good approximation of what a universal turing machine; and the applications we use can be treated as specific Turing machines running on the universal Turing machine.

2 DNA

In 1953 a young American geneticist named James Watson (1928–) and his British colleague Francis Crick (1916–2004) had a brilliant insight into the building plan of the DNA molecule that accomplished several things:

- It demonstrated that DNA almost certainly contained the hereditary material of cells and organisms;
- It revealed how cells could copy DNA to pass it along to their offspring; and
- It showed how the molecule might change through mutations, which make evolution possible.

2.1 DNA structure

A single strand of DNA can be likened to a storage tape that supports a four symbol alphabet, where $\Gamma = \{A, G, C, T\}$ representing the nucleotides adenine, guanine, cytosine and thymine. These nucleotides are not bounded to each other directly but rather hang from a phosphate and sugar backbone. The DNA strand's backbone has a polarity; a sequence of DNA is distinct from its reverse. In order to represent the polarity we name the ends of a piece of DNA according to the structure of its phosphate backbone. One end is called the 5′ end —its terminal phosphate is attached to the 5′ oxygen of a sugar andd the other end end is called the 3′ end — its terminal is attached to the 3′ oxygen of the sugar.

Taken as pairs the nucleotides A and T, and the nucleotides G and C are said to be complementary. This means that an A-T pair or a C-G pair can form weak, non-covalent bonds known as hydrogen bonds that serve to hold them together. When a stretch of single-stranded DNA encounters a stretch of another single-stranded DNA that has a complementary sequence, the hydrogen bond interactions between complementary pairs join the two strands in a process called annealing. A piece of DNA will only anneal to its complement if it has the opposite polarity.

2.2 DNA as a computing medium

DNA looks like a tape of a Turing Machine. The similarity has prompted computer scientists to think of it as a media for computation, and the similarity is very attractive for the following reasons:

1) DNA is the genetic material
   DNA is the storage medium for genes—the plans for the protein molecular machines that perform most of the chemistry in all living things. Our genes determine our morphology, and influence our behaviour. And we are interested in modifying genes in a general way (e.g. Turing machines).
2) There are many enzyme-mediated chemical reactions for DNA.
3) DNA is small and easily copied.
   There are 67 atoms per A-T pair and 66 per C-G pair. DNA supports symbols so this gives capacity of 1 bit per 33 atoms for double-stranded DNA. The average molecular weight of one base pair is 660 Daltons. This gives 0.33 kg DNA / mole bits.
4) Reactions between DNA species at equilibrium are completely reversible.
   Charles Bennet proposed that computers based on DNA would be good candidates for practical reversible computers. Normally computations carried out in irreversible steps lose information and dissipate heat. Because enzyme catalyzed operations between DNA species at equilibrium are completely reversible any computations that we embed in them are reversible as well.

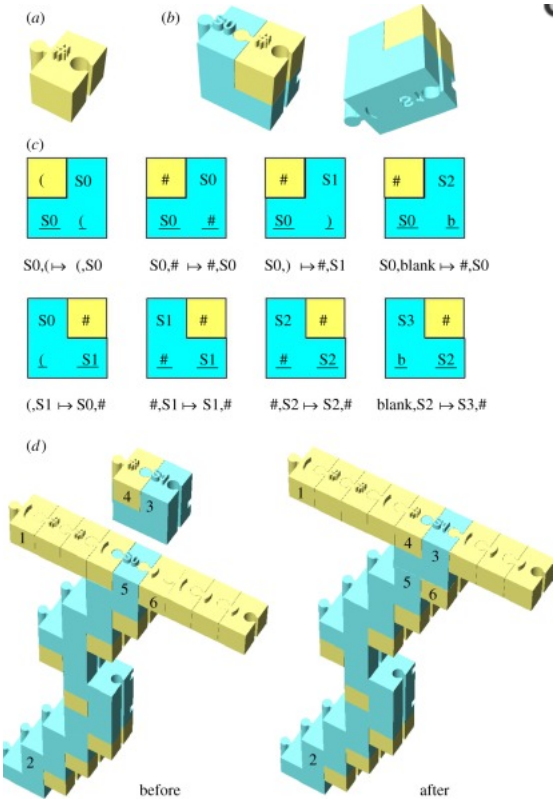
3 Ribosome

The ribosome is a complex molecular machine, that serves as the site of biological protein synthesis. Ribosomes link amino acids together in the order specified by messenger RNA (mRNA) molecules. Ribosomes consist of two major components: the small ribosomal subunit, which reads the RNA, and the large subunit, which joins amino acids to form a polypeptide chain.

Adelman concluded his seminal paper by saying: 'In the future, research in molecular biology may provide improved techniques for manipulating macromolecules. Research in chemistry may allow for the development of synthetic designer enzymes. One can imagine the eventual emergence of a general-purpose computer consisting of nothing more than a single macromolecule conjugated to a ribosome-like collection of enzymes that act on it'. Here, we attempt to advance this vision by proposing a detailed logical design for such a computer, with the ultimate goal of constructing a general-purpose programmable computer that can operate in vivo and interact with its biochemical environment. As the tools of molecular biology and chemistry are insufficient at present to realize this design with biomolecules, we realized it in a working mechanical implementation. This mechanical device serves as a proof-of-concept of the logical design as well as a high-level operational specification for a biomolecular implementation.

In 1994, Adleman showed how to compute using DNA molecules and standard molecular biology laboratory techniques. Adleman's method involves encoding combinatorial search problems with DNA sequences, and using in vitro selection techniques to synthesize and isolate DNA sequences that encode solutions to these problems. Subsequent works further developed and expanded this research direction.

The mechanical computer employs a chain of basic building blocks referred to as alphabet monomers, to represent the Turing machine's tape, and uses another set of building blocks eferred to as transition molecules, to encode the machine's transition rules. The transition encoding is similar to a Wang tile construction, which is also at the basis of DNA computing via self-assembly, and also to the concept of modified tRNA proposed as part of a ribosome-like computing device. A transition molecule loaded with an alphabet monomer specifies a computational step of the computer similar to the way an aminoacyl-tRNA specifies a translation step of the ribosome. The set of loaded transition molecules constitutes the computer's program.



The computer operates on two chains of building blocks simultaneously. One chain, referred to as the tape polymer, represents the Turing machine's tape and is edited by the computer similar to the way a Turing machine modifies its tape. The other chain, referred to as the trace polymer, is a by-product of the computation constructed incrementally from displaced transition molecules and displaced alphabet monomers, and has no analogue in the theoretical Turing machine. A transition molecule, referred to as the active transition molecule, joins the two polymers. The active transition molecule is embedded in the tape polymer and represents the location of the Turing machine's read/write head as well as the machine's internal state. At the same time, the active transition molecule is the terminal molecule of the trace polymer, representing the most recent transition of the computation. (Note that in this design, the read/write head is located between adjacent tape cells, not on a specific cell, unlike a standard Turing machine; this change does not affect the computational capabilities of the machine.)

The computer is made of two subunits, referred to as small and large, each with a tunnel called the small tunnel and the large tunnel, respectively. The small tunnel provides incoming loaded transition molecules with access to the active transition molecule and to its adjacent alphabet monomer. Access is controlled by gating mechanisms which block transition molecules that are ill-formed or do not match the current state and current tape symbol. These mechanical analogues of allosteric conformational changes open the channel only when a valid incoming transition molecule approaches. The large tunnel holds the active transition molecule and the tail of the trace polymer being constructed.

The computer operates in cycles, processing one transition molecule per cycle. In each cycle, an incoming loaded transition molecule that matches the current state and its adjacent alphabet monomer becomes the new active transition molecule and its accompanying alphabet monomer is incorporated into the tape polymer. This is achieved by displacing the currently active transition molecule and the matched alphabet monomer, effectively editing the tape polymer, and elongating the trace polymer by the displaced molecules. Specifically, when processing a left transition molecule the computer moves left to accommodate the molecule, if necessary, and displaces the currently active transition molecule and the alphabet monomer to its left by the new molecule. The computer processes a right transition molecule similarly by moving right and displacing the alphabet monomer to the right of the active transition molecule. The theoretical Turing machine has an infinite tape, with only a finite portion of it being non-blank at any point during the computation. For obvious reasons, and in line with natural information representation by biopolymers, the mechanical Turing machine represents the two infinite blank portions of the tape implicitly. A special mechanism, shown in figure 2, detects the left and right ends of tape and treats each as a blank symbol. Special left blank transition molecules detect if the state they specify is at the left end of the tape and if so write a symbol and move to the left by activating this mechanism. Right blank transition molecules achieve the symmetric effect. The size of the mechanical Turing machine and its components do not make it susceptible to Brownian motion. Hence, assembly of transition molecules, pushing transition molecules down the small tunnel, and moving the small as well as large subunits relative to each other and relative to the tape polymer, all need to be carried out manually. The small as well as large units are designed and connected so that the small unit can wobble one symbol to the left or to the right relative to the large unit. In the left position, the current state and the symbol to its left are exposed to incoming left move transitions. Similarly, in the right position, right-move transitions may take effect. A peculiar aspect of the design is that this non-deterministic wobble precedes and enables the application of a corresponding move transition, and the transition taken has the effect of moving the wobble range one symbol in its direction (left or right). The computer is designed to be robust to Brownian motion in that only a transition which matches the current state and symbol can release the levers that would allow it to take effect.

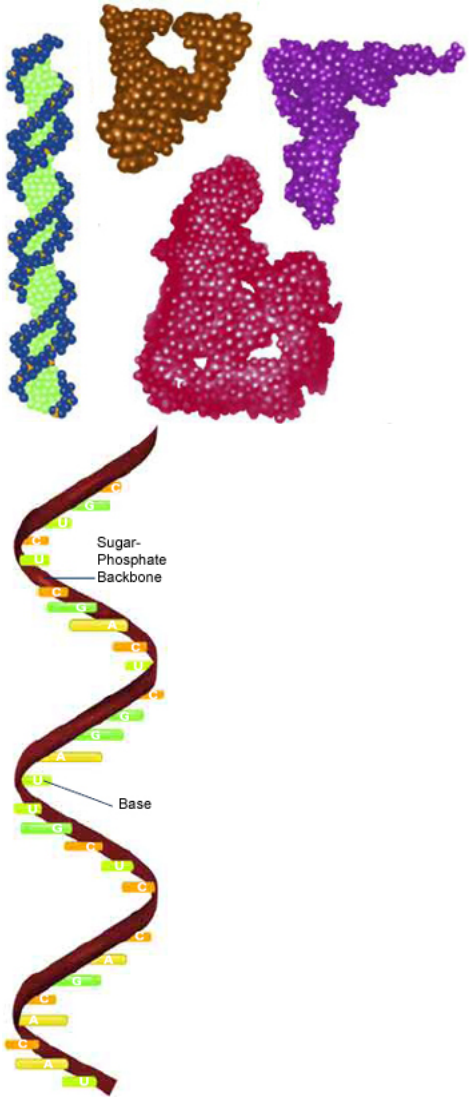
When considering a future biomolecular realization of the mechanical Turing machine, one must realize that the device was designed to operate on three-dimensional building blocks by applying mechanical analogues of polymer elongation, cleavage and ligation, movement along a polymer, and control by molecular recognition unleashing allosteric conformational changes. Logically, the device is not more complicated than biomolecular machines of the living cell, and all its operations are part of the standard repertoire of these machines; hence, a biomolecular embodiment of the device is not infeasible. Specifically, a transition can be effected through the Brownian motion of an applicable loaded transition molecule into the tunnel of the small unit, followed by molecular recognition between the current state and symbol, and the state and symbol of the loaded transition molecule that triggers an allosteric conformation change. The conformational change in turn enables the

incorporation of the new state and symbol instead of the old state and symbol, presumably through two cleavages and two ligations of the tape polymer.

The mechanical computer is similar to the ribosome in several other respects. Both operate on two polymers simultaneously, and their basic cycle consists of processing an incoming molecule that matches the currently held molecules on the first polymer, elongating the second polymer and moving sideways. Like the ribosome in the living cell, the computer requires supporting devices similar in function to aminoacyl-tRNA synthetases to load bare transition molecules with correct alphabet monomers, and a device similar in function to proteases to decompose the trace polymer and make its components available for reuse. However, unlike the ribosome, which only 'reads' the messenger RNA in one direction, the computer edits the tape polymer and may move in either direction.

4 RNA

RNA stands for ribonucleic acid. It is an important molecule with long chains of nucleotides. A nucleotide contains a nitrogenous base, a ribose sugar, and a phosphate. Just like DNA, RNA is vital for living beings.



DNA is defined as a nucleic acid that contains the genetic instructions used in the development and functioning of all known living organisms. RNA molecules are involved in protein synthesis and sometimes in the transmission of genetic information.

However unlike DNA, RNA comes in a variety of shapes and types. While DNA looks like a double helix and a twisted ladder, RNA may be of more than one type. RNA is usually single-stranded, while DNA is usually double-stranded. In addition, RNA contains ribose while DNA contains deoxyribose. Deoxyribose lacks one oxygen atom. RNA has the bases Adenine (A), Uracil (U) (instead of thymine in DNA), Cytosine (C) and Guanine (G).

Deoxyribose sugar in DNA is less reactive because of C-H bonds. DNA is stable in alkaline conditions. DNA has smaller grooves where the damaging enzyme can attach which makes it harder for the enzyme to attack DNA.

The main job of RNA is to transfer the genetic code need for the creation of proteins from the nucleus to the ribosome. This process prevents the DNA from having to leave the nucleus. This keeps the DNA and genetic code protected from damage. Without RNA, proteins could never be made.

5 Amino Acids

Amino acids are organic compounds containing amine (-NH2) and carboxyl (-COOH) functional groups, along with a side chain (R group) specific to each amino acid. The key elements of an amino acid are carbon, hydrogen, oxygen, and nitrogen, although other elements are found in the side chains of certain amino acids. About 500 amino acids are known (though only 20 appear in the genetic code) and can be classified in many ways.[4] They can be classified according to the core structural functional groups' locations as alpha- (α-), beta- (β-), gamma- (γ-) or delta- (δ-) amino acids; other categories relate to polarity, pH level, and side chain group type (aliphatic, acyclic, aromatic, containing hydroxyl or sulfur, etc.). In the form of proteins, amino acid residues form the second-largest component (water is the largest) of human muscles and other tissues. Beyond their role as residues in proteins, amino acids participate in a number of processes such as neurotransmitter transport and biosynthesis.

6 DNA condon table

As we have already mentioned a Turing machine has an instruction set (or table), a DNA condon table is just like that. The genetic code is traditionally represented as an RNA codon table because, when proteins are made in a cell by ribosomes, it is mRNA that directs protein synthesis. The mRNA sequence is determined by the sequence of genomic DNA.

1st base	2nd base								3rd base
	T		C		A		G		
T	TTT	(Phe/F) Phenylalanine	TCT	(Ser/S) Serine	TAT	(Tyr/Y) Tyrosine	TGT	(Cys/C) Cysteine	T
	TTC		TCC		TAC		TGC		C
	TTA	(Leu/L) Leucine	TCA		TAA[B]	Stop (*Ochre*)	TGA[B]	Stop (*Opal*)	A
	TTG		TCG		TAG[B]	Stop (*Amber*)	TGG	(Trp/W) Tryptophan	G
C	CTT		CCT	(Pro/P) Proline	CAT	(His/H) Histidine	CGT	(Arg/R) Arginine	T
	CTC		CCC		CAC		CGC		C
	CTA		CCA		CAA	(Gln/Q) Glutamine	CGA		A
	CTG		CCG		CAG		CGG		G
A	ATT	(Ile/I) Isoleucine	ACT	(Thr/T) Threonine	AAT	(Asn/N) Asparagine	AGT	(Ser/S) Serine	T
	ATC		ACC		AAC		AGC		C
	ATA		ACA		AAA	(Lys/K) Lysine	AGA	(Arg/R) Arginine	A
	ATG[A]	(Met/M) Methionine	ACG		AAG		AGG		G
G	GTT	(Val/V) Valine	GCT	(Ala/A) Alanine	GAT	(Asp/D) Aspartic acid	GGT	(Gly/G) Glycine	T
	GTC		GCC		GAC		GGC		C
	GTA		GCA		GAA	(Glu/E) Glutamic acid	GGA		A
	GTG		GCG		GAG		GGG		G

,

7 Proteins

Proteins are large, complex molecules that play many critical roles in the body. They do most of the work in cells and are required for the structure, function, and regulation of the body's tissues and organs.

Proteins are made up of hundreds or thousands of smaller units called amino acids, which are attached to one another in long chains. There are 20 different types of amino acids that can be combined to make a protein. The sequence of amino acids determines each protein's unique 3-dimensional structure and its specific function.

Examples of functions done by proteins are the following:

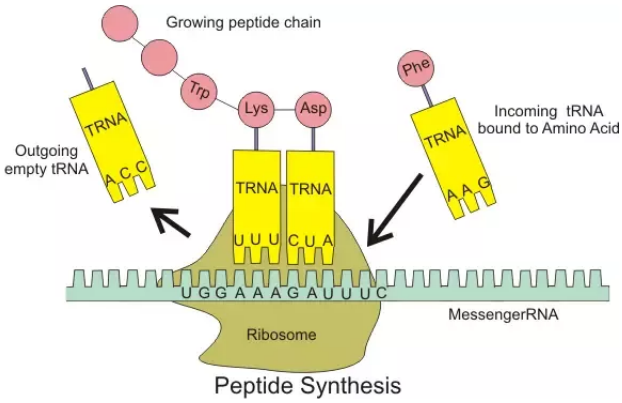
Function	Description
Antibody	Antibodies bind to specific foreign particles, such as viruses and bacteria, to help protect the body.
Enzyme	Enzymes carry out almost all of the thousands of chemical reactions that take place in cells. They also assist with the formation of new molecules by reading the genetic information stored in DNA.
Messenger	Messenger proteins, such as some types of hormones, transmit signals to coordinate biological processes between different cells, tissues, and organs.

8 Molecular Turing Machine

When comparing a Turing machine with a biological cell, you can say that they are very similar, as we have already discussed, in the following ways:

- **DNA** would be the **tape** of a Turing machine.
- **Ribosome** would be the **head**.
- **RNA** would be the **state register**.
- **Amino acids** would be the **states**.
- **DNA condon table** would be the **instruction table**.
- **Proteins** would be the **output tape**.

Protein synthesis is the process whereby biological cells generate new proteins; it is balanced by the loss of cellular proteins via degradation or export. Translation, the assembly of amino acids by ribosomes, is an essential part of the biosynthetic pathway, along with generation of messenger RNA (mRNA), aminoacylation of transfer RNA (tRNA), co-translational transport, and post-translational modification. Protein biosynthesis is strictly regulated at multiple steps. They are principally during transcription (phenomena of RNA synthesis from DNA template) and translation (phenomena of amino acid assembly from RNA).



References

[1] Turing A. M. 1937. On computable numbers, with an application to the entscheidungsproblem. Proc. Lond. Math. Soc.

[2] Aspray W. 1989. John von Neumann's contributions to computing and computer science. IEEE Ann. Hist. Comput.

[3] Adleman L. M. 1994. Molecular cAdar R., Benenson Y., Linshiz G., Rosner A., Tishby N., Shapiro E. 2004. Stochastic computing with biomolecular automata. omputation of solutions to combinatorial problems.

[4] Lipton R. J. 1995. DNA solution of hard computational problems. Science 268, 542–54510.1126/science.7725098

[5] Aandreaa S., Jervell. A Universal Turing Machine, Computer Science Logic.

[6] Adleman L. M. On Constructing a Molecular Computer, Draft. 1995

[7] Bennet, C.H. The Thermodynamics of Computation — a Review.

[8] Douglas S. M., Bachelet I., Church G. M. 2012. A ogic-gated nanorobot for targeted transport of molecular payloads.

[9] Simmel F. C. 2007. Towards biomedical applications for nucleic acid nanodevices

[10] Xie Z., Wroblewska L., Prochazka L., Weiss R., Benenson Y. 2011. Multi-input RNAi-based logic circuit for identification of specific cancer cells.

[11] Basu S., Gerchman Y., Collins C. H., Arnold F. H., Weiss R. 2005. A synthetic multicellular system for programmed pattern formation

[12] Ramaswamy S., et al. 2001. Multiclass cancer diagnosis using tumor gene expression signatures. Proc. Natl Acad. Sci.

[13] Rothemund P. W. 2006. Folding DNA to create nanoscale shapes and patterns.

[14] Stojanovic M. N., Mitchell T. E., Stefanovic D. 2002. Deoxyribozyme-based logic gates

[15] Kimball, John. "The Genetic Code"

[16] Hovemeyer, David. CS 340 lecture notes.