Computing vs. Genetics

J. M. Barreiro
Universidad Politécnica de Madrid, Spain

J. Pazos
Universidad Politécnica de Madrid, Spain

ABSTRACT

This chapter first presents the interrelations between computing and genetics, which both are based on information and, particularly, self-reproducing artificial systems. It goes on to examine genetic code from a computational viewpoint. This raises a number of important questions about genetic code. These questions are stated in the form of an as yet unpublished working hypothesis. This hypothesis suggests that many genetic alterations are caused by the last base of certain codons. If this conclusive hypothesis were to be confirmed through experiementation if would be a significant advance for treating many genetic diseases.

INTRODUCTION

The mutual, two-way relationships between genetics and computing (see Table 1) go back a long way and are more wide-ranging, closer and deeper than what they might appear to be at first sight. The best-known contribution of genetics to computing is perhaps evolutionary computation. Evolutionary computation's most noteworthy representatives are genetic algorithms and genetic programs as search strategies. The most outstanding inputs from computing to genetics are reproductive automata and genetic code deciphering. Therefore, section 2 will deal with von Neumann reproductive automata. Section 3 will discuss genetic code. Section 4 will introduce the well-know χ^2 test because of this importance in establishing the working hypothesis. Later, section, will address genome deciphering. And finally section 6 will establish the conjecture or working hypothesis, which is the central conclusion of the paper, and define the future research lines.

SELF-REPRODUCING AUTOMATA

The most spectacular contribution of computing to genetics was unquestionably John von Neumann's

premonitory theory of self-reproducing automata, i.e. the construction of formal models of automata capable of self-reproduction. Von Neumann gave a conference in 1948 titled "The General and Logical Theory of Automata" (Von Neumann, 1951, 1963) establishing the principles of how a machine could self-reproduce. The procedure von Neumann suggested was at first considered an interesting logical and mathematical speculation more than anything else. However, von Neumann's view of how living beings reproduced (abstractedly simpler than what it might appear) was acclaimed five years later, when it was confirmed, after James D. Watson and Francis Harry C. Crick (1953(a)) discovered the model of DNA.

It was as of 1950s that Information Theory (IT) exercised a remarkable influence on biology, as it did, incidentally, on many other fields removed from the strictly mathematical domain. It was precisely as of then that many of the life sciences started to adopt concepts proper to IT. All the information required for the genesis and development of the life of organisms is actually located in the sequence of the bases of long DNA chains. Their instructions are coded according to a four-letter alphabet A, T, C and G. A text composed of the words written with these four letters constitutes the genetic information of each living being. The Nobel prize-winning physicist Erwin Schrödinger (1944) conjectured the existence of genetic code, which was demonstrated nine years later by Watson and Crick (1953(a), (b)), both awarded the Nobel prize for this discovery. It was in the interim, in 1948, when von Neumann established how a machine could self-reproduce.

Table 1. Computing vs. genetics

From genetics to computing	**From computing to genetics**
Natural Computation (NC) ≡ Evolutionary Computation (EC) [Genetics Algorithms (GA) + Evolution Strategies (ES) + Evolutionary Programming (EP)] + Neural Networks (NN) + Genetic Programming	1940 Claude Elwood Shannon (1940) defended his PhD thesis titled "An Algebra for Theoretical Genetics".
1966 Fogel, Owens and Walsh (1966) establish how finite state automata can be evolved by means of unit transformations and two genetic operators: selection and mutation.	1944 Erwin Schrödinger (1983) conjectured that genetic code existed.
1973 Rechemberg (1973) defined the evolutionary strategies of finite state machine populations.	1948 John Von Neumann (1966) established the principles underlying a self-reproducing machine.
1974 Holland (1975) and disciples defined genetic algorithms.	1953 Crick (Watson, 1953) luckily but mistakenly named the small dictionary that shows the relationship between the four DNA bases and the 20 amino acids that are the letters of protein language genetic code.
1992 Koza (1992) proposed the use of the evolutionary computation technique to find the best procedure for solving problems, which was the root of genetic programming.	1955 John G. Kemeny (1955) defined the characteristics of machine reproduction and how it could take place.
1994 Michalewitz (1992) established evolutionary programs as a way of naturally representing genetic algorithms and context-sensitive genetic operators.	1975 Roger and Lionel S. Penrose (Penrose, 1974) tackled the mechanical problems of self-reproduction based on Homer Jacobson's and Kemeny's work.
	1982 Tipler (1982) justified the use of self-reproducing automata.

Self-Reproduction

In his utopian novel "Erewhon", which is the mirror image of "nowhere", Samuel Butler (1982) explores the possibility of machines using men as intermediaries for building new machines. There have been many examples of machines built by other machines in the last century. Steam engines were used to build other steam engines, and machine tools made all sorts of devices. Long before the computer era, there were mechanical and electrical machines that formed metal to build engines. The Industrial Revolution was largely possible thanks to machine tools: machines that were conceived exclusively to build other machines. However, devices that can be used to make other devices were not exactly the type of machine tools that the English Prime Minister, Benjamin Disraeli, had in mind when he said, "The mystery of mysteries is to view machines making machines", i.e. machines building other machines without human involvement. In other words, machines that self-reproduce or, if you prefer a less controversial term, are self-replicate.

What is the meaning of the word reproduction? As John G. Kemeny (1955) pointed out, if reproduction is understood as the creation of an object that is identical to the original one from nothing, it is evident that a machine cannot self-reproduce, but neither could a human being. For reproduction not to violate the principle of energy conservation, some raw material is required. What characterises the reproduction of life is that the living organism is capable of creating new, similar organisms from the inert matter in its surroundings. If we accept that machines are not alive and we insist on the fact that the creation of life is a fundamental characteristic of reproduction, the problem is settled: a machine is incapable of self-reproduction. The problem is therefore reformulated so as not to logically rule out the reproduction of machines. To do this, we need to omit the word "living". So, we will stipulate that the machine should be capable of creating a similar new organism from simple components existing in its surroundings.

Scientists have been dreaming about creating machines programmed to produce replicas of themselves since 1948. These replicas would produce other replicas and so on, without any limit whatsoever. The theory that established the principles of how such a feat could be achieved was first formulated in 1948. This theory has two aspects, which could be termed logical and mechanical. The mechanical question was addressed, among others by Lionel S. and Roger Penrose (1974) and will not be considered here. The logical part, which is our concern here, was first researched by von Neumann at the Institute for Advanced Study in Princeton. It was there that von Neumann suggested the possibility of building a device that had the property of self-reproduction. The method involved building another describable machine from which it followed logically that this machine would carry a sort of tail that would include the code describing how to reproduce the body of the machine and how to reproduce the actual code. According to Kemeny (1955), a colleague of von Neumann, the basic body of the machine would be composed of a box containing the constituent parts, to which a *tail* would be added that stored the units of information. From the mechanical viewpoint, it was considered that the elementary parts from which the machine would have to be built would be rolls of tape, pencils, rubbers, empty tubes, quadrants, photoelectric cells, motors, batteries and other similar devices. The machine would assemble these parts from the surrounding raw material, which it would organise and transform into a reproduction of itself. As von Neumann's aim was to solve the logical conditions of the problem, the incredible material complications of the problem were left aside for the time being.

Von Neumann's proposal for building machines that have the reproductive capability of the living organisms was originally considered as an interesting mathematical speculation more

than anything else, especially taking into account that computers back then were 30 or more tonne giants and were little more than devices for rapidly performing mathematical operations. How could we get a machine to produce a copy of itself? A command from a human programmer to "Reproduce!" would be out of the question, as the machine could only respond "I cannot self-reproduce because I don't know who I am". This approach would be as absurd as if a man gave his partner a series of bottles and glass flasks and told her to have a child. In von Neumann's opinion, any human programmer proposing to create a dynasty of machines would have to take the following three simple actions:

1. Give the machine a full description of itself.
2. Give the machine a second description of itself, which would be a machine that has already received the first description.
3. Finally, order the machine to create another machine that is an exact copy of the machine in the second description and order the first machine to copy and pass on this final order to the second machine.

The most remarkable thing about this logical procedure is that, apart from being simpler than it may appear, it was von Neumann's view of how living creatures reproduce. A few years after his conference, his ideas were confirmed when the biologists Crick and Watson (1953 (a) and (b)) found the key to genetic code and discovered the secret of organic reproduction. It was essentially the same as the sequence for machine reproduction that von Neumann had proposed. In living beings, deoxyribonucleic acid (DNA) plays the role of the first machine. The DNA gives instructions to ribonucleic acid (RNA) to build proteins; RNA is like DNA's "assistant". Whereas the RNA performs the boring task of building proteins for its parent organisms and offspring, the DNA plays the brilliant and imaginative role of programming its genes, which, in the case of a human baby, will decide whether it has blonde or brown hair and whether it will be of an excitable or calm temperament. In short, DNA and RNA together carry out all the tasks that the first von Neumann machine has to perform to create the second machine of the dynasty. And, therefore, if we decide to build self-reproductive machines, there is important biological evidence that von Neumann came across the right procedure to do so a long time ago.

But, one might wonder, why would anyone want to build computers that make copies of themselves? The procedure could at best be bothersome. Suppose that someone went to bed after having spent the evening working at his computer and, when he woke up the next day found that there were two computers instead of one. What would these regenerating computers be useful for? The answer is that they will be used at remote sites to perform difficult and dangerous tasks that people cannot do easily. Consequently, we have to consider at length the possible location of such places. What is it that is holding back human biological development? Why, over thirty-five years after man first set foot on the moon, is there still no permanent lunar colony? Over three quarters of a century have passed since man first managed to fly and most human beings are still obliged to live on the surface of the Earth. Why? The astronomer Tipler (1980), from the University of California in Berkeley, answered this question very clearly when stated that it was the delay in computer not rocket technology that was preventing the human race from exploring the Galaxy. It is in space, not on Earth, where the super intelligent self-reproducing machines will pay off, and it is in space where the long-term future of humanity lies. It is fascinating to consider how Tipler and others who have studied the far-off future consider how von Neumann machines will make it possible, firstly to colonise the solar system of planets and then the Milky Way with over 150,000 million suns.

Of course, none of the machines mentioned here have, as far as we know, been built, but there is nothing to stop them from being built. Scientifically, nothing stands in the way of their construction. Whether it is technologically feasible to make replicas is another question, and a tricky one at that.

GENETIC CODE

Code vs. Cipher

Technically, a code is defined as a substitution at the level of words or sentences, whereas a cipher is defined as a substitution at the level of letters. Ciphering means concealing a message using a cipher, whereas coding means concealing a message using a code. Similarly, the terms decipher is applied to the discovery of a ciphered message, i.e., in cipher, whereas the term decode is applied to the discovery of a coded message. As we can see, the terms code and decode are more general, and are related to both codes and ciphers. Therefore, these two terms should not be confused through misuse. For example, Morse code, which translates the letters of the English alphabet and some punctuation marks to dots and dashes, is not a code in the sense of a form of cryptography because it does not conceal a message. The dots and dashes are simply a more convenient form of representing the letters for telegraphy. Really, Morse code is nothing other than an alternative alphabet.

Code is formally defined as follows. Let A* be a free monoid (Hu, 1965) engendered by the set A, i.e. A* is the set of finite-length words formed by means of concatenation (the associative law of internal composition) with the symbols of A and with a neutral element, namely, the empty word. A code $C = \{c_1,...,c_i\}$, then, is a subset of A*, where the elements of c_i are the words of the code and n_i denotes the size of the word c_i.

A code is said to be binary, ternary or, generally, n-ary when A is formed by, respectively, two, three or, generally, n symbols. If all the words are of the same length, C is said to be a fixed-length or block code. Otherwise, C is said to be a variable-length code.

Let σ be the alphabet of a source of information, which is defined as the source that emits symbols of an effect $\sigma \{s_1,...,s_5\}$, whose probability of appearance is given by $\pi_i (1 \le i \le r)$. Then a coding is an application φ of σ in C, which is extended to an application φ^* of σ^* in C*. And, of course, decoding is an application ψ of C* in σ^*. A code has only one decoding, i.e. φ^* is injective, and ψ or ψ^* is the identity.

Genetic Code

The expression genetic code is now used to mean two very different things. The lay public uses it often to refer to the full genetic message of the organism. Molecular biologists use it to allude to the small dictionary that shows how to relate the four-letter language of the nucleic acids to the twenty-letter language of the proteins in the same way as Morse code relates the dots-and-dashes language to the twenty-six letters of the alphabet. Here we will use the term in this sense. However, the technical term for such a rule of translation is not, strictly speaking, *code* but cipher, just as Morse code should be called Morse cipher. This Crick did not know at the time, which was a stroke of luck, as *genetic code* sounds much better than *genetic cipher*. In actual fact, Crick correctly referred to the set of bases contained in the chromosomes as ciphered text or key, but added that the term *key* or ciphered text was too limited. The chromosomal structures are at the same time the instruments that develop what they foresee. They represent both the legal text and the executive power or, to use another comparison, they are both the architect's plans and the builder's workforce.

Code is the core of molecular biology just like the periodic table of elements is at the heart of

chemistry, but there is a profound and transcendental difference. The periodic table is almost certainly valid and true all over the universe. However, if there is life in other worlds, and if this life is based on nucleic acids and proteins, of which there is no guarantee, it is very likely that the code there would be substantially different. There are even small code variations in some terrestrial organisms. Genetic code, like life itself, is not an aspect of the eternal nature of things, but, at least partly, product of an accident.

The central lemma of genetic code is the relationship between the sequence of the bases of DNA or of its transcribed m-RNA and the sequence of protein amino acids. This means that the sequence of bases of a gene is collinear with the sequence of amino acids of its product polypeptide. What makes this code, which is the same in all living beings, so marvellous is its simplicity. A set of three bases or codon specifies an amino acid. The t-RNA molecules, which work as protein synthesis adaptors, read the codons sequentially.

Crick et al.'s 1961 experiments (Crick et al., 1961) established that genetic code had the following characteristics:

1. The alphabet: A1 = {A,G,C,T}.
2. Coding relationship. A group of three bases codes an amino acid. This group of three bases is called, as mentioned above, codon or triplet.
3. Non-optimality: The fact that there is a code is such is likely to be due to structural, electrochemical and physical criteria applied to the molecules involved in the described processes. The optimal base is actually 3 (Pazos, 2000).
4. Non-overlapping: In a code of non-overlapping triplets, each codon specifies only one amino acid, whereas in a code of overlapping triplets, ABC specifies the first amino acid, BCD the second, CDE the third and so on. Studies of the amino acid sequence in the protein cover of mutants of the tobacco mosaic virus indicated that normally only one amino acid altered, leading to the conclusion that genetic code does not overlap.
5. Codon reading sequentiality: The sequence of bases is read sequentially from a fixed starting point. There are no commas, i.e. genetic code does not require any punctuation or signal whatsoever to indicate the end of a codon or the start of the next one. There is only an "initiation signal" in the RNA, which indicates where the reading should start. This signal is the AUG codon that codes the amino acid methionine.
6. Code inefficiency or degeneracy: There is more than one *word* or coding codon for most amino acids. Mathematically, this means that there is a *superjective* application between the codons and the amino acids plus the chain initiation and termination signals, which is transcendental for the subject of this paper. Table 2 shows genetic code. Only tryptophan and methionine are encoded by a single triplet. Two or more triplets encode the other eighteen. Leucine, arginine and serine are specified by six codons each. However, under normal physiological conditions, the code is not ambiguous: each codon designates a single amino acid. From this table, the respective amino acid can be located, given the position of the bases in a codon, as follows. Suppose we have the m-RNA codon 5'AUG 3', then we start at A in Table 2, then go to U and then to G and we find methionine.

Figure 1 shows a finite-state automaton that recognises the DNA alphabet and translates the codons into amino acids.

7. Importance of the bases in the codon: The codons that specify the same amino acid are called synonyms, e.g. CAU and CAC are synonyms for histidine. Note that the synonyms are not distributed arbitrarily in

Table 2. An amino acid specified by two or more synonyms occupies only one cell, unless there are over four synonyms. The amino acids in each cell are specified by codons whose first two bases are the same but differ as to the third, e.g. GUU, GUC, GUA and GUG. Most synonyms differ only as to the last base of the triplet.

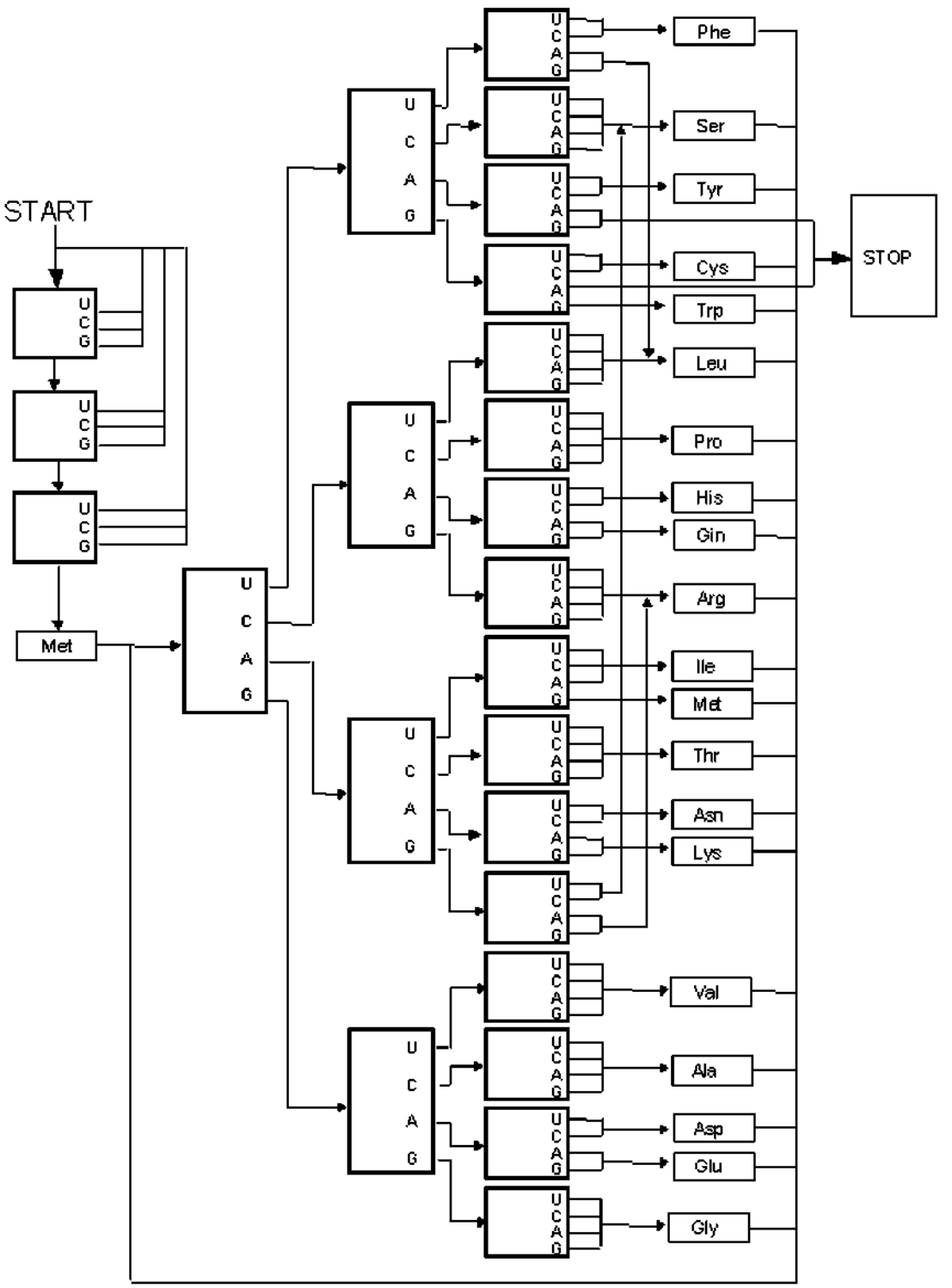
Looking at the code, we find that XYC and XYU always code the same amino acid, whereas XYG and XYA almost always do. It is clear then that the first two letters of each codon are significant factors, whereas the third appears to be less important and not to fit in as accurately as the others. The structural basis for these equivalences is evident owing to the nature of the anticodons of the t-RNA molecules. However, and this is what we want to stress in this paper, the last base of the codon is fundamental from the functional viewpoint, as it is the one that will characterise the behaviour of the protein and the original gene in which it is located. The generalised degeneracy of genetic code has two biological implications. On the one hand, degeneracy reduces the noxious effects of mutations to a minimum. One plausible reason why code is degenerated is that redundancy provides a safety mechanism. But utility is also a possible ground: it is quite plausible that the silent mutations may have long-term benefits. On the other hand, code degeneracy can also be significant insofar as it allows DNA to largely modify its base composition without altering the sequence of amino acids it encodes. The content of [G] + [C] could encode the same proteins if they simultaneously used different synonyms.

8. Termination signals: the last characteristic of the code refers to the fact that of the 64 possible triplets, there are three that do not encode any amino acid, and these are UAG, UAA and UGA.

Table 2. Genetic code

Position 1	**Position 2**				**Position 3**
(5' end)	U	C	A	G	**(3' end)**
U	Phenylalanine	Serine	Tyrosine	Cysteine	U
	Phenylalanine	Serine	Tyrosine	Cysteine	C
	Leucine	Serine	Stop	Stop	A
	Leucine	Serine	Stop	Tryptophan	G
C	Leucine	Proline	Histidine	Arginine	U
	Leucine	Proline	Histidine	Arginine	C
	Leucine	Proline	Glutamine	Arginine	A
	Leucine	Proline	Glutamine	Arginine	G
A	Isoleucine	Threonine	Asparagine	Ser	U
	Isoleucine	Threonine	Asparagine	Ser	C
	Isoleucine	Threonine	Lysine	Arginine	A
	Met (Start)	Threonine	Lysine	Arginine	G
G	Valine	Alanine	Aspartic acid	Glycine	U
	Valine	Alanine	Aspartic acid	Glycine	C
	Valine	Alanine	Glutamic acid	Glycine	A
	Valine	Alanine	Glutamic acid	Glycine	G

Figure 1. A finite-state automaton that translates genetic code to amino acids



χ^2 TEST

Suppose that we find that a series of events E_1, E_2, …, E_j occur in a given sample with frequencies o_1, o_2, …, o_j, called "observed frequencies", and that, according to the rules of probability, they would be expected to occur with frequencies e_1, e_2, …, e_j, called theoretical or expected frequencies, as shown in Table 3.

A measure of the discrepancy between the observed and expected frequencies is given by the χ^2 statistic, as

$$\chi^2 = \frac{(o_1 - e_1)^2}{e_1} + \frac{(o_2 - e_2)^2}{e_2} + \ldots\ldots\ldots + \frac{(o_j - e_j)^2}{e_j} = \sum_{i=1}^{j} \frac{(o_j - e_j)^2}{e_j}$$

where, if the total frequency is N, $\Sigma o_i = \Sigma e_j = N$.

If $\chi^2 = 0$, the observed and expected frequencies are exactly equal, whereas if $\chi^2 > 0$, they are not. The greater the value of χ^2, the greater the discrepancies between the two frequencies are.

The χ^2 sample distribution is very closely approximated to the chi-square distribution given by

$$Y = Y_0 (\chi^2)^{1/2}(\nu - 2)\, e^{-1/2}\chi^2 = Y_0 \chi^{\nu-2} e^{1/2}\chi^2$$

where ν is the number of degrees of freedom given by:

a. ν = k-1, if the expected frequencies can be calculated without having to estimate population parameters from the sample statistics.
b. ν = k-1-m, if the expected frequencies can only be calculated by estimating m parameters of the population from the sample statistics.

In practice, the expected frequencies are calculated according to a null hypothesis H_0. If, according to this hypothesis, the calculated value of χ^2 is greater than any critical value, such as $\chi^2_{0.95}$ or $\chi^2_{0.99}$, which are the critical values at the significance levels of 0.05 and 0.01, respectively, it is deduced that the observed frequencies differ significantly from the expected frequencies and the H_0 is rejected at the respective significance level. Otherwise, the H_0 will be accepted or, at least, not rejected.

Now, looking at Table 4 taken from Jack Lester King and Thomas H. Jukes (1969), we ran the χ^2 test on these data and found that the value of χ^2 is 477.809. As this value is much greater than the expected $\chi^2_{0.95}$, which, for the 19 degrees of freedom in respect of the twenty amino acids, is 38.6, we find that the observed frequencies differ very significantly from the expected values, thereby rejecting the H_0. Accordingly, it is to be expected that the bases are not associated as triplets at random and, therefore, an explanation needs to be sought.

DECIPHERING THE GENOME

As already mentioned, genes are usually located in the chromosomes: cellular structures whose main component is deoxyribonucleic acid, abbreviated to DNA. DNA is formed by complementary chains, made up of long sequences of nucleotide units. Each nucleotide contains one

Table 3. Sample with associated frequencies

Event	E1	E2	…	Ej
Observed frequency	o1	o2	…	oj
Expected frequency	e1	e2	…	ej

Table 4. Amino acids and triplets associated with their frequencies according to King and Jukes (1969)

AMINO ACIDS	Triplets	Number of appearances	Observed frequency (%)	Expected number	Expected frequency (%)	$\frac{(O_i - O_i)^2}{e_j}$
Serine	UCU.UCA.UCC UCG.AGU.AGC	443	8.0	472	8.6	1.782
Leucine	CUU.CUA.CUC CUG.UUA.UUG	417	7.6	434	7.9	0.666
Arginine	CGU.CGA.CGC CGG.AGA.AGG	230	4.1	582	10.6	212.9
Glycine	GGU.GGA.GGC GCG	406	7.4	390	7.1	0.656
Alanine	GCU.GCA.GCC GCG	406	7.4	395	7.2	0.306
Valine	GUU.GUA.GUC CUG	373	6.8	330	6.0	5.603
Threonine	ACU.ACA.ACC ACG	340	6.2	373	6.8	2.919
Proline	CCU.CCA.CCC CCG	275	5.0	275	5.0	0
Isoleucine	AUU.AUA.AUC	209	3.8	280	5.1	18.0
Lysine	AAA.AAG	395	7.2	302	5.5	28.639
Glutamic acid	GAA.GAG	318	5.8	258	4.7	13.953
Aspartic acid	GAU.GAC	324	5.9	192	3.5	90.750
Phenylalanine	UUU.UUC	220	4.0	121	2.2	81
Asparagine	AAU.AAC	242	4.4	225	4.1	1.284
Glutamine	CAA.CAG	203	3.7	210	3.8	0.233
Tyrosine	UAU.UAC	181	3.3	165	3.0	1.551
Cysteine	UGU.UGC	181	3.3	137	2.5	14.131
Histidine	CAU.CAC	159	2.9	164	3.0	0.152
Methionine	AUG	99	1.8	99	1.8	0
Tryptophan	UGG	71	1.3	88	1.6	3.284
TOTAL		**5,492**	**100.0**	**5,492**	**100.0**	$\chi^2 = 477.809$

of the four possible nitrogenised bases: adenine (A), cytosine (C), thymine (T) and guanine (G), which only associate in two possible ways with each other: A with T and C with G. Usually, some portions of DNA form genes and others do not. In the case of human beings, the portions that are genes make up only approximately 10% of total DNA. The remainder appears to have nothing to do with protein synthesis; it is, until it finds a functionality, genetic trash so to speak. In any case, the reading and interpretation of this set of symbols that make up DNA can be compared to deciphering the hieroglyphics of life. If Jean-François Champollion's deciphering of hieroglyphic

script from the Rosetta Stone was arduous and difficult, imagine deciphering $3x10^9$ symbols from a four-letter alphabet. To give an idea of the magnitude of the endeavour, suffice it to say, for example, that the sequence of DNA nucleotides written would take up a space equivalent to 150 volumes similar to the telephone book of a city like Madrid, with four million inhabitants. Or, to establish a more illustrative comparison, if a virus gene that has 3,000 pairs of bases takes up one page of a book composed of 3,000 letters and a gene of a bacterium, which contains three million pairs of bases, would be equivalent to a 1,000-page book, the human genome, composed of three thousand million bases, would take up a library of a thousand books. Perhaps one form of deciphering the hieroglyphics would be what Eric Steven Lander of the MIT proposed when he said that just as the organisation of the chemical elements in the periodic table lent coherence to a mass of previously unrelated data, we will end up discovering that the tens of thousands of genes of existing organisms are made of combinations of a much smaller number of simpler genetic modules or elements or, so to speak, primordial genes. The important point is that human genes should not be considered as completely different from each other. Rather, they should be seen as families that live all sorts of lives. Having completed the table, structural genetics will leave the field open to functional genetics or the practical use of the table. For example, the difference between two people boils down to no more than 1% of the bases. Most genes have only two, three or four variants, that is, some 300,000 principal variants.

Going back now to deciphering the genome, sequencing the genome refers to solely stating the bases A, T, C and G of the genome, i.e. reading without understanding. Therefore, it is to spell out or, at most, babble the genome. For the time being, genome sequencing simply involves determining how the thousands of millions of bases of which it is composed are chained. As Daniel Cohen (1994) said, who we follow in this section, it is not hard to realise how arid the task is, as it involves examining a text that more or less reads as follows:

TCATCGTCGGCTAGCTCATTCGACCATCG-TATGCATCACTATTACTGATCTTG...,

and goes on for millions of lines and thousands of pages. Of course, it was to be expected that a language that has a four-letter alphabet would be much more monotonous than contemporary languages, whose Latin, Cyrillic alphabets, etc., are composed of over twenty letters.

But sequencing the genome is not the last stop, as it has to be deciphered, i.e. its meaning has to be understood like learning to read letters and converting them into ideas. And this is the tricky thing. There is, in actual fact, no way of foreseeing when, how and to what extent it will be possible to decipher the sequenced genome. It would be marvellous, but not very realistic, to suppose that learning to read the genome would unexpectedly lead to its understanding just as children learn the letters and suddenly cross some mysterious threshold and start to understand what they are reading. Reality is, however, much tougher, and the deciphering of the genome looks like a very hard and thankless task. This is due not so much to the poverty of the alphabet but to the ignorance of the language.

Just to give an illustrative example of what we have just said, try to read, albeit for no more than twenty minutes, and aloud (no cheating!), a novel written in a language you don't know (Turkish, Serbo-Croatian, etc.) transliterated to the twenty-six letter Latin alphabet and see what a headache you get. Suppose now that you have no choice, because you are on a desert island and the only book you have is written in one of these languages and you do not have dictionary on hand. Therefore, you would have to make do with what little you know and be patient and perseverant. If

you do this, you will end up becoming acquainted with some features of the unknown language. For example, you will identify recurrent patterns, establish analogies, discover some rules, other meanings, interesting similarities, etc., etc., etc. The first thing we find is that the genetic language has a peculiarity, which, at least in principle, is disconcerting: it does not just consist of sequences furnished with a precise meaning, the sequences of interest which are called genes. It all includes jumbled paragraphs situated both between gene and gene, intergenes, and inside the genes, intragenes, which divide the meaningful sequence. To date, no one has been able to find out what all this filling is for. And, what is even more exasperating is that these extravagant series of letters make up over ninety per cent of the genome, at least, the human genome, which is an interminable list of genes and intergenes with non-coding intragenes situated within the very genes.

Now consider a volume of poetry by Quevedo (1995), but written in an unknown language and according to the following genomic style, although for the readers' comfort, the text has been translated into Spanish:

kvlseimkmifdsqmoieurniñaedpvaeighmlkehimdogcoleizoapglkchdjthzkeivauozierdmofmoimthaaoekkkkkkkghdmorsleidjtldifltsfifesithgmelimlkajchwchmqurozqdaverirlmeoarusndorkejmtsimeormtlehoekdmzriglalmethoslerthrosazlckthmekromlsdigquemelthnlslrejtestalkhlrjjsletehrejrozthiolalelyrletuolgdhnartlgldrtlmalkjfsdñanaioemzlekthldiimsekjrmlkthomsdlkgldheozkelldesgyrureiotpleghkdssseruieormdkfjhjkddddgoghfjdsenbvcxxwqsd ...

Biologists are now beginning to distinguish the characteristic sequences that initiate and terminate the words of the genome and can better identify genes from what are not genes. Therefore, with a bit of training, geneticists will manage to pick out, from the above mess, the following sentence:

Quitar codicia no añadir dinero hace a los hombres ricos Casimiro

But, even after identifying the words of the poem, they would still be a long way away from imagining what Casimiro, to whom the poem is addressed, was like.

Returning to the analogy with the human language, the task facing those who are sequencing the genome is to make an inventory of the thousands of words of the biological dictionary of humanity. Only the inventory. The explanation of the words will come afterwards. To give an idea of the endeavour, this dictionary, at a rate of thirty genes and approximately a million characters per page, will have three thousand pages, given that the genome has three thousand million bases. The equivalent, in pages and weight, of the two-volume Diccionario de la Real Academia de la Lengua Española.

But, unlike usual dictionaries, the words will be arranged not in alphabetical but in chromosomal order. Instead of being grouped in the respective sections A to Z, the words of the human genomic dictionary would be arranged in 23 chapters, one for each pair of chromosomes duly numbered from 1 to 23 in the conventional order attributed to them by biologists. The genes of the first pair are the longest, whereas those of the twenty-first and twenty-second are the shortest. These are, as mentioned above, the chromosomes termed "autosomes", i.e. non sexual, from the Greek *soma*, meaning body. The twenty-third pair is the sex chromosomes X and Y.

So, in the first chapter, a chromosome of x million bases would take up x pages at a rate of a million characters per page, in the second, y..., and in the twenty-first, z. "Beaconing" and mapping the genome is equivalent to paginating the dictionary; ordering the pages from 1, 2, 3, ... to three thousand, in 23 chapters, yet without filling the pages in. At this stage, we still do not know what there is on each of the pages, except for a thousand words, set out here and there. Having

done the pagination or, if you prefer, map-making, the experts will be able to start on the systematic sequencing of the genes, i.e., fill the pages with words still without explanation.

Then, in a few years time, when we have the sequence of the whole human genome, we will see that page 1 of chapter 1 contains the words, or their genomic equivalent, not in alphabetical order (a, aal, aam, Aaronic, ..., ab, aba, abaca), but arranged as stipulated by nature through the chromosomes (e.g., grace, disgrace, graceless, followed by a word, like "everyday", which bears no resemblance to its foregoer, then gracecup, graced, followed by an "incongruous" plum and division and, then, disgraceful, aggrace, begrace, etc.). Looking at these words more carefully, we find that many are composed of the letters "grace" and we wonder what this common factor means. Further on, we find that the common factor "gen" appears in genetic and also, in another place in another chapter, in genesis, genius, ingenious, genial, etc. All the "grace" and all the "gen" can then be entered into a computer to discover how often they appear and fully examine the sequences in which they are accommodated. The whole thing can then be reclassified taking note of the significant statistical data. On other occasions, researchers may come across a series of the style: complain, complainant, complaint, plaint, plaintive, plaintiff, and may be fortunate enough to know the meaning of its common factor, in this case, "plaint", because it is associated with some hereditary particular or other that has already been studied at length. They will then be able to venture to try out new rules of construction, new combinations, taking the same common root. This is how the cartouches helped to decipher hieroglyphics. In other words, genes, like the words of the language can be grouped by families, the words of different human languages that revolve around the same concept share the same root. Likewise, it is to be supposed that the genes whose sequence is similar fulfil similar functions. As a matter of fact, it now appears that genes that resemble each other come from one and the same ancestral gene and form families of genes with similar functions. These are termed multigenic families, whose genes may be disseminated across several chromosomes. Often, neighbouring genes have a similar spelling even if they do not start with the same letters. And homonymic genes are often synonyms, genes that are written differently but have similar meanings. This will mean that they can be used like *puntales* (a fragment of plain text associated with a fragment of ciphered text) were used to decipher secret codes.

The analysis will be gradually refined and fine-tuned. Finally, we will know how to distinguish the equivalents of linguistic synonyms. By detecting the common roots, we will even learn to trace back the genealogy of certain genes, i.e. follow their evolutionary lineage. Complex biological functions like breathing, digestion or reproduction are assimilated to sentences whose words are inscribed on different pages of the genomic dictionary. Now, if there is a first book, the genetic dictionary written according to the topographical order of the chromosomes, evolution has written another thousand after learning this dictionary, on physiology, growth, ageing, immunity, etc., to the book of thought, which, doubtless, will never be finished. It is even conceivable that one and the same gene could acquire a different meaning depending on its position in one or other genetic sentence, just as a word in human language depends on the context. Additionally, biological functions have been invented as evolution has become more complex. It is likely that genetic combination was at the same time enriched by new rules integrating earlier levels in as many other Russian dolls of *algorithms*, placed inside each other in a subtle hierarchy.

In sum, the syntax and style of the genetic language has gradually been refined, and it remains for us to discover its semantics and pragmatics. The messages that determine eye colour, skin texture or muscular mass are doubtless not the same as those that induce the immune system, cellular

differentiation or cerebral *wiring*. Obviously, many fundamental concepts of this language are unknown. Even after sequencing is complete, there will still be a lot to research to do and it will take perhaps centuries of work to get things straight, if we ever do. The question is that DNA is neither an open book nor a videotape.

FUTURE TRENDS

Now already numerous, wide-ranging and rewarding, the interrelations between genetics and computing will expand increasingly in the future. The major trends that are likely to be the most constructive are:

A) Biological computation. Under this label, DNA computing deserves a mention. While the early work on DNA computing dates back to Adelman, there is still a long way to go before we can build what is now being referred to as the *chemical universal Turing machine* (CUMT). Its construction would have an extraordinary effect on the understanding of genetics and computer science, as it would allow theoretical and experimental approaches and models in both fields to be combined holistically.

B) DNA, the brain and computers have one thing in common: all three process information. Consequently, it is evident that, at a given level of abstraction, their operational principles will be the same. This will lead, sooner or later, to the discovery of an information theory that accounts for the behaviour of the three processors and even more. In actual fact, the world has been considered so far as being formed by matter and energy, both of which are, since Einstein's famous formula of human destiny, $E=\pm mc^2$, equivalent. Now, to understand today's world (both at the macroscopic level, for which the general theory of relativity accounts, i.e. black holes, and the microscopic level accounted for by quantum physics, i.e. Wheeler's delayed choice experiment), information needs to be added to the matter-energy equation. Now, this theory would of course encompass Shannon's communication theory, but would go further than Shannon's premise does. It might perhaps only retain his notions of the information unit "bit" and negative entropy. One of the authors is already working in this field and expects to have some preliminary results to report in a few months' time.

C) Genetics-computation hybridization. The exchange of approaches between genetics and computation will provide a hybrid form of dealing with problems in both fields. This will improve problem solving in both domains. For example, geneticists will be able to routinely apply concepts commonly used in computing, like abstraction or recursiveness. This way they will acquire profound skills for solving complex problems. Additionally, the huge quantities of information that DNA employs to develop its full potential, as well as the complexity of its workings, will be excellent guides for dealing with problems in the world of computing.

CONCLUSION AN FUTURE RESEARCH LINES

The classical scientific dogma, which is or should be inculcated to any university student, is that first conjectures or working hypotheses are formulated and are then tested. But, of course, to formulate such conjectures or hypotheses, the facts, as such, need to be taken into account, and these facts are:

a) Proteins owe their function to their structure or folding, i.e. to their shape, which depends on the order of the amino acid sequence of

which they are composed. And this order is again determined by the sequence of the DNA bases.

b) From Table 4, taken from King and Jukes (King, 1969), we calculated the χ^2 and found that the χ^2 test results offer no doubt as to the fact that the distribution of the triplets and their translation to amino acids is not due to chance, quite the opposite.

c) According to genetic code, we know that several triplets yield the same amino acid.

This leads us to formulate the proposed working hypothesis or conjecture:

An individual's *situation* will depend on what triplet and in what position it yields a particular amino acid.

Testing:

To test this conjecture, we have to, and this is what we are in the process of doing, take the following steps.

S1. Determine a genetic disease of unique aetiology

S2. Try to associate the possibilities of "triplets versus generated amino acids" relationships within the gene causing the disease and establish, if possible, a causal relationship and, if not, a correlation. For this purpose, we have to define the right sample size.

S3. If a causality or correlation greater than 0.8 is established, go to S1 with a new case. If the number of cases is greater than the proposed sample size and causality or correlation was established for all cases, accept the hypothesis, if not, reject it.

Of course, when genetic code has been completely deciphered, this type of hypotheses will make no sense, because the DNA will explain its message and its resultant consequences. But, in the meantime, it is a way of understanding why some diseases occur. This will, of course, lead to its prevention, cure or, at least, to the relief of its effects, through genetic engineering.

REFERENCES

Butler, S. (1982). *Erewhon*. Barcelona: Bruguera.

Cohen, D. (1994). *Los Genes de la Esperanza*. Barcelona: Seix Barral.

Crick, F.H.G.; Barnett, L.; Brenner, S. & Watts-Tobin, R.J. (1961) General Nature of the Genetic Code for Proteins. *Nature*, 192, 1277-1232.

Fogel, L. & Atmar, J.W (Eds.) (1992) *Proceedings First Annual Conference on Evolutionary Programming.*

Holland, J. H. (1975) *Adaptation in Natural and Artificial Systems*. Ann Arbor, Michigan: University Michigan Press.

Hu, S-T. (1965). *Elements of Modern Algebre*. San Francisco, Ca: Holden-Day, Inc.

Kemeny, J. G. (1955). Man Viewed as a Machine. *Scientific American*, 192 (4), *58-67.*

King, J. L. & Jukes, T. H. (1969). Non Darwinian Evolution. *Science*, 164, 788-798.

Koza, J.R. (1992). *Genetic Programming*. Reading, MA: The MIT Press.

Michalewitz, Z. (1992). *Genetic Algorithms + Data Structures = Evolutionary Programs*. New York: Springer Verlag.

Pazos, J. (2000). El Criptoanálisis en el Desciframiento del Código Genético Humano. In J. Dorado et al. (Eds.), *Protección y Seguridad de la Información* (pp. 267-351). Santiago de Compostela: Fundación Alfredo Brañas.

Penrose, L. J. (1974). Máquinas que se Autorreproducen. In R. Canap, et al., *Matematicas en*

las Ciencias del Comportamiento (pp: 270-289). Madrid: Alianza Editorial, S.A.

Quevedo, F. de. (1995). *Poesía Completa I* (pp: 43-44). Madrid: Turner Libros, S.A..

Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution*. Stuttgart, Germany: Fromman-Holzboog Verland.

Schrödinger, E. (1944). *What is Life?* Cambridge: Cambridge Universities Press. Spanish translation (1983). *¿Qué es la Vida?* (pp: 40-42). Barcelona: Tusquets Editores, S.A.

Tipler, F. J. (1980). Extraterrestrial Intelligent Beings Do Not Exist. *Quarterly Journal of the Royal Astronomical Society,* 21, 267-281.

Tipler, F. J. (1982). We Are Alone In Our Galaxy, *New Scientist*, 7 October, 33-35.

Von Neumann, J. (1951). La Teoría Lógica y General de los Autómatas. In L.A. Jeffres (Ed), *Cerebral Mechanisms in Behaviour* (pp. 1-41). New York: John Willey and Sons.. And in, J. von Neumann (1963). *Collected Works* (pp 288-328). Oxford: Pergamon.

Von Neumann, J. (1966). *Theory of Self-Reproducing Automata*. Urbana. Illinois: Illinois University Press.

Watson, J. D. & Crick, F. H. C. (1953 a). Molecular Structure of Nucleic Acid. A Structure for Deoxyribose Nucleic Acid. *Nature* 171, 737-738.

Watson, J. D. & Crick, F. H. C. (1953 b). Genetic Implications of the Structure of Deoxyribonucleic Acid. *Nature* 171, 964-967.

ADDITIONAL READING

Adleman L. M. (1998). Computing with DNA. *Scientific American Magazine* nº 279 pp. 54-61.

Adleman L. M. (1994). Molecular computation of solutions to combinatorial problems. *American Association for the Advancement of Science* Vol. 266, Issue 11 pp. 102-1024 Washington, DC, USA Univ. of Southern California, LA.

Bekenstein J. D. (2003) La información en el Universo Holográfico. *Investigación y ciencia*, nº 325, pp. 36-51.

Benenson Y, Gil B, Ben-Dor U, Adar R, Shapiro E. (2004). An Autonomous Molecular Computer for Logical Control of Gene Expression. *Nature*, 429:423–429.

Copeland, B. J. y Proudfoot, D. (1999). Un Alan Turing desconocido. Prensa Científica S.A. *Investigación y Ciencia* nº 273.

Feynman, R.P. (1960). There's Plenty of Room at the Bottom: An Invitation to enter a New World of Physics. *Engineering and Science* 10:23 (5) pp. 23-36.

Gifford D. K. (1994). On the Path to Computation with DNA. in Science, Vol. 266, pp. 993–994.

Gray J, Liu DT, Nieto-Santisteban M, Szalay AS, De Witt D, Heber G. (2005). Scientific Data Management in the Coming Decade. *Technical Report MSR-TR*-2005-10. Microsoft Research.

Griffiths, A.J.F.; Gelbart, W.M.; Miller, J.H. and Lewontin, R.C. (2002). *Genética Moderna* 1ª Edición en español. McGraw-Hill/Interamericana.

Gruska J. (1999). *Quantum Computing*. McGraw-Hill.

Kurzweil R. (2006). When Computers Take Over. Nature Publishing Group. *Nature*, Books and Arts, vol 437(440) pp. 421-422 23.

Lehn J-M. (2004). Supramolecular Chemistry: from Molecular Information Towards Selforganization and Complex Matter. *Reports on Progress In Physics* 67:249-265.

Lipton R. J. (1995). DNA solution of Hard Computational Problems. *Science*, 268: 542-545.

Lloyd S. (2002). Computational Capacity of the Universe. *Physical Review Letters* 10Volume 88, nº 23.

Lloyd S. (2000). Ultimate Physical Limits to Computation. *Nature* 406 Aug 31; 406(6799): 10 47-54.

Martin-Vide C.; Paun G.; Pazos J.; Rodríguez-Paton A. (2003) Tissue P systems. Elsevier. *Theoretical Computer Science*, Vol. 296, Number 2, 8, pp. 295-326(32).

Mount, D.W. (2004). *Bioinformatics: Sequence and Genome Análisis*. Edition: 2nd edition, Cold Spring Harbor Laboratory Press.

Nielsen M. A. and Chuang I. L. (2007). *Quantum Computation and Quantum Information*. Cambridge University Press. Cambridge UK.

Regev A., Shapiro E. (2002). Cellular Abstractions: Cells as Computation. *Nature*, 419: 343.

Russell, P.J. (2002). *iGenetics*. Benjamin Cummings. USA

Wing J. (2006). Computational Thinking. *Communications of the ACM*. Vol. 49, No. 3, pp. 33-35.

Wolfram S.A. (2002). *New Kind of Science*. Champaign, IL:Wolfram Media Inc.

Zauner K-P. (2005). Molecular Information Technology. *Critical Reviews in Solid State and Material Sciences*, Volume 30, nº 1, pp. 33-69(37).