Crick's Hypothesis Revisited: The Existence of a Universal Coding Frame

Jean-Louis Lassez*, Ryan A. Rossi
Computer Science Department, Coastal Carolina University
jlassez@coastal.edu, raross@coastal.edu

Axel E. Bernal
Computer Science Department, University of Pennsylvania
abernal@seas.upenn.edu

Abstract

In 1957 Crick hypothesized that the genetic code was a comma free code. This property would imply the existence of a universal coding frame and make the set of coding sequences a locally testable language. As the link between nucleotides and amino acids became better understood, it appeared clearly that the genetic code was not comma free. Crick then adopted a radically different hypothesis: the "frozen accident". However, the notions of comma free codes and locally testable languages are now playing a role in DNA Computing, while circular codes have been found as subsets of the genetic code. We revisit Crick's 1957 hypothesis in that context. We show that coding sequences from a wide variety of genes from the three domains, eukaryotes, prokaryotes and archaea, have a property of testable by fragments, which is an adaptation of the notion of local testability to DNA sequences. These results support the existence of a universal coding frame, as the frame of a coding sequence can be determined from one of its fragments, independently from the gene or the organism the coding sequence comes from.

1. Introduction

In the early stages of the discovery of the genetic code, Crick hypothesized that the genetic code's structure was endowed with specific information theoretic properties [1].

This would be not only most satisfying intellectually speaking, but would also help explain the extraordinary fact that the genetic code is essentially the same for all organisms. This property would imply that the frame of a fragment of a coding region from any gene or organism can be determined independently from the start or stop codons and independently from the gene or the organism it comes from. We refer to this as the universal frame property of coding regions.•

When the mapping from codons to amino acids was better understood and the genetic code appeared to be not comma free, Crick abandoned his early hypothesis to adopt a radically opposite one, the "frozen accident": the structure of the genetic code and its uniqueness are due to an accident in evolution rather than being due to its functionality from an information theoretic point of view. As a consequence, at present, the frame is determined by careful statistical analysis taking into account the specific origin of the organism. In an interesting historical record [2], Crick's notion of comma free code was credited as being the prettiest wrong idea in all of the 20th century science. However, researchers still pursue, along different lines, the hypothesis that the genetic code's structure is not accidental [3], and in the exciting and growing area of DNA computing, researchers are not dealing with "accidental" codes, they build them according to good information theoretic properties, and comma free codes are again under study [4]. Furthermore, significant results have been found regarding evolutionary aspects of the genetic code [5,6] as well as new techniques to detect the coding regions and the coding frames [7,8,9], when the notion of comma free code is replaced by the more appropriate notion of circular code [10,11].

In this work, we revisit Crick's hypothesis and reformulate it with the hindsight of 50 years of progress in biology, formal languages and coding theories. We argue that the appropriate notion to study the structure of the genetic code and coding sequences is not the notion of comma free or the notion of circular code, but the notion of testable by fragment, which we introduce as a variant of the notion of locally testable [12], as it is more suitable to the analysis of genomic sequences.

• Work supported by NSF Grant ATM-0521002

21st International Conference on
Advanced Information Networking and Applications Workshops (AINAW'07)
0-7695-2847-3/07 $20.00 © 2007 IEEE

To this effect, we show that from a single arbitrarily chosen gene, DKEYP-117E10.6, from the zebra-fish, we can infer by similarity the coding frame of 95% of 2939 genes in the three domains, prokaryotes, eukaryotes and archaea. We then show how one can infer the coding frame of a gene from a fragment of its coding sequence with a certain probability as a function of the length of the segment and independently of the genome or isochore to which the gene belongs. We demonstrate how these results support the existence of a universal coding frame by using a relaxed version of Crick's hypothesis, in which more than one codon is needed to retrieve the coding frame. We also stress the significant role played by the partitioning of the genetic code into three subsets, the T codes [5,6], related to early evolutionary models of the genetic code [13, 14].

In the conclusion we mention research directions that arise from our studies. We believe in particular that the methods we introduced can be used for the analysis of other genomic features, such as pseudo genes, gene complements, and UTR's.

Further information and results can be found at: http://cs.coastal.edu/ucf/

2. Comma Free to Testable by Fragment

Here we present the motivations and intuitions that lead to the reformulation of Crick's hypothesis.

From a formal language/coding theory point of view (but not necessarily from a biological point of view), one can think of the coding sequence of a gene as a sequence of words written in the alphabet {A,C,G,T} having special properties. Each word is translated into a symbol representing an amino acid. There is no special symbol separating the words. What should be the properties of this set of words?

In order to avoid ambiguities in translation we do not want to have a set of words such as: {AC, TGAC, ACTG} because the message ACTGAC could be parsed in different ways: AC/TGAC and ACTG/AC, leading to an ambiguity: which of the corresponding translations is the intended one? As a consequence we need a set of words that forms a code, that is a set of words such that any message can be parsed into code words in a unique way, leading to a unique possible translation. If all words have the same length the problem is solved trivially because there is a unique way to parse messages from such a code. However, a form of ambiguity still exists: Consider the code {ACT, TAG, CTT, AGA}, the sequence ACTTAG can be parsed unambiguously into ACT/TAG, so we know that the words ACT and TAG will be those to translate. But if the sequence is extracted from a longer sequence whose extremities are unknown, say ……ACTTAG….. then we cannot be sure, because the subsequence CTT is also a code word and could be a candidate for translation depending on the frame. What is worse, assuming that the intended frame is the one that corresponds to …ACT/TAG…, an error in transmission that would drop the first letter (A) would cause a frame-shift CTT/AG…. and a non intended translation.

Crick's early hypothesis was that the genetic code is comma free; in that case it ought to have a very strong property: no single trinucleotide in a frame-shift can be translated, because no trinucleotide in a frameshift belongs to the code. As a consequence, the occurrence of a single code word in a coding region defines the frame, regardless of the start or stop codons, the gene or the organism it comes from. Hence, Crick's hypothesis can be reinterpreted as claiming the existence of a universal frame, with the comma free property as means of establishing this hypothesis.

We now know of course, that the notion of comma free is far too drastic, indeed it implies that we can determine the frame given any fragment of length 5 in the coding region; still, that does not necessarily rule out the existence of a universal frame. We first relax the notion of comma free codes. In [11] the definition of parasite sub-messages was introduced. If {ACT, TAG, CTT, AGA} is the code and ACTTAG the intended message, then CTT is a parasite sub-message. A comma free code is a code without parasites. A code with bounded parasitism allows parasite sub-messages of at most length d code words. If the code has bounded parasitism we need to see a sequence of d+1 code words in order to determine the frame. So if our goal is to test the universal frame hypothesis it is reasonable to consider such codes rather than the most restrictive comma free.

We now consider another way of addressing the universal frame hypothesis: local testability [12]. Informally, if we can decide that a sequence belongs to a language L by analyzing independently all its factors of a given length, the language L is called locally testable. It is easy to create examples of locally testable languages; for example, consider the set L of sequences that do not contain the subsequence ATA, testing all sub-words of length 3 in the sequence for equality to ATA allows us to determine if the sequence belongs to L. A great example of "something" not locally testable is provided by Escher, who was followed by a number of (creative) imitators in MAD magazine, with their drawings of "impossible" objects. Look at his famous "endless staircase" (image to be found on the site http://cs.coastal.edu/ucf/), if there is a

window that allows us to see only four steps at a time, each view is compatible with a regular staircase. But when you have global view of the whole, you realize it is not a staircase. This conflict between local and global has been used systematically in a number of Escher's other drawings. We know that (finite) codes with bounded parasitism generate sets of messages that are locally testable and conversely [11].

However if we are interested in the universal frame hypothesis, the fact that the codes are comma free, or have bounded parasitism is only of secondary importance if the set of messages is locally testable. Indeed local testability may allow us to find the frame, even if the underlying code does not have the above-mentioned properties. The reason is simple: there could be rules that restrict the generation of parasite sub-sequences, for instance rules that restrict long repeats of AAA, CCC, GGG and TTT. We then still should be able to verify the universal frame hypothesis despite the eventual lack of properties of the underlying genetic code.

More formally let G be a code and G* the set of all messages that it can generate and let L, strict subset of G* a language defined by some grammatical rules. The definition of the frame of words in L could very well come from the rules rather than from the properties of the code G. Therefore, in order to verify the universal frame hypothesis, we can relax Crick's hypothesis from G comma free to G being a code with bounded parasitism, to G* being locally testable, to L being locally testable. All these notions are very closely related in a formal way, described in the next section; however it is by using the most appropriate one that the problem's solution will become apparent. In that respect we will consider two further adaptations of the mathematical formalism to our situation. In coding theory as well as in formal language theory, two words are considered different if they are not syntactically identical. This is too strict for our purpose; we will use the notion of similarity between words rather than identity. Furthermore, the formal definition of a locally testable language is far more restrictive than what its intuitive and informal motivation infers. We will then use a more appropriate variant of this formal definition that is still very much in the spirit of the informal one. For this reason, we will not use the terminology "locally testable" but instead the terminology "testable by fragments". We now can reformulate Crick's hypothesis: What is the length of the shortest fragment of coding sequence, if it exists, that will allow us to determine the frame, independently of the gene or the organism it comes from?

3. Preliminary Definitions and Results

A set of words S is a code if and only if any message, that is any word of S*, can be parsed in a unique way into words of S.

The results we give now can be derived from well known more general theorems [10,11]. However we are in a situation in which they can be established in a simple and intuitive way, when we *only consider codes X whose words have the same length k.*

Let ***m*** be a message from a code **X**, that is a sequence of words from the code **X**. If a subsequence ***p*** of **X**, in a shifted frame, is also made of words of **X**, ***p*** is called a *parasite sub-message* and ***m*** will be referred to as the *intended message.*

A *comma free* code is a code that does not admit any parasite sub-messages. As a consequence the frame is determined by any occurrence of a code word in a message.

Remark 1. The genetic code is not comma free as any sequence of length 3 in a gene is a code word, regardless of the frame in which it occurs.

A code **X** has *bounded parasitism* of degree *d* if there are parasite sub-messages in words of **X*** made of at most *d* words of **X**.

A code has *spread parasitism* if one can find messages with parasite sub-messages of arbitrary length.

As a consequence we have:

Proposition 1. A code **X** has bounded parasitism of degree *d* if and only if the code $\mathbf{X}^{d+1}$ is comma free.

Hence, if the **X** code has bounded parasitism of degree *d*, any occurrence of a sequence of *d*+1 words of **X** determines the frame.

We will relate these notions to the concept of locally testable. The set **X*** of messages from a code **X** is *strictly locally testable* if and only if we can decide if a word w belongs to **X*** in the following way: there exists a number *d* such that the prefix of *w* of length *kd* is in **X***, as well as the suffix of *w* of length *kd*, and all factors of *w* of length *kd* are factors of words of **X***.

In other words we can decide if *w* is a message from **X** by sliding a window of a given length along *w* and independently analyse the properties of each window.

Theorem 1. A Comma Free code **X** generates a set of messages **X*** which is strictly locally testable.

Proof (informal). Let w be a word of **X*** it is straightforward to see that it satisfies the conditions. What we have to show is that if *w* does not belong to **X***, then some condition will not be satisfied. First case, *w*'s length is not a multiple of *k*. Assuming that all the other conditions are met, the suffix of *w* of length 2*k* cannot satisfy the condition because it would imply that a word of **X** appears in a shifted frame, in contradiction with the fact that **X** is comma free. Second case, *w* is of length multiple of *k*. Then one of the *k*-uples in the coding frame does not belong to **X**. This will be found immediately if it is one of the first two. Assume it is the third. Then the sequence made of the second and the third triplets cannot be a factor of words of **X*** because the third triplet does not belong to **X**, and if it was a shifted factor it would imply that **X** is not comma free. Now if the third triplet belongs to **X** we can shift the argument to the next triplet and repeat the argument.

Theorem 2. If **X*** is strictly locally testable then X has bounded parasitism

Proof (informal). If **X** does not have bounded parasitism, then we can have parasite sub-messages of arbitrary length. We can then make a word that does not belong to **X***, but has arbitrarily long prefixes and suffixes that belong to **X***. As a consequence windows of fixed size cannot discriminate between the two competing frames and the word w will be accepted as a word of **X***.

So we have established the links between bounded parasitism, comma free and local testability, we will now briefly mention how circular codes [11] are related. Circular codes have applications in dynamical systems, coding and automata theory, combinatorics [15], and more recently in theoretical biology [5,6,7,8,9], as we will point out in the next section. They are of relevance here because in the finite case they are identical to codes with bounded parasitism, and it is this property that has been used in the applications in biology, not the circularity. The "circularity" aspect of circular codes might be more relevant in DNA computing where one computes with plasmids [16].

4. T-representations and Similarities

The following circular codes (which are in fact codes with bounded parasitism) have been found as subsets of the genetic code [5]:

$\mathbf{X_0}$ = {AAC, AAT, ACC, ATC, ATT, CAG, CTC, CTG, GAA, GAC, GAG, GAT, GCC, GGC, GGT, GTA, GTC, GTT, TAC, TTC}

$\mathbf{X_1}$ = {ACA, ATA, CCA, TCA, TTA, AGC, TCC, TGC, AAG, ACG, AGG, ATG, CCG, GCG, GTG, TAG, TCG, TTG, ACT, TCT}

$\mathbf{X_2}$ = {CAA, TAA, CAC, CAT, TAT, GCA, CCT, GCT, AGA, CGA, GGA, TGA, CGC, CGG, TGG, AGT, CGT, TGT, CTA, CTT}

These codes have remarkable properties, and have been used to help identify coding regions for prokaryotes and eukaryotes [7]; other circular codes have been found for archaea [8], and yet others are used to find the frame in bacterial coding regions [9]. But it is the codes $T_0=X_0 U \{AAA, TTT\}$, $T_1=X_1 U \{CCC\}$ and $T_2=X_2 U \{GGG\}$ that we will consider as their union forms the whole genetic code. These codes [5,6] when translated in the two letter genetic alphabet {R,Y} (R = purine, that is A or G, while Y = pyrimidine, that is C or T) allow to retrieve a codon model for primitive protein coding genes [13,14].

The issue is then the analysis of the distribution of these codes in genes. For this purpose, we associate three T-Representations to any coding sequence u:

The first representation, T, is obtained by replacing each codon by 0 if it belongs to T_0, 1 if it belongs to T_1 and 2 if it belongs to T_2. This representation corresponds to the coding frame, while the two others represent the shifted frames. The second representation T^+ is obtained by elimination of the first letter of u and applying the preceding construction. Finally, the third representation T^{++} is obtained by eliminating a second letter from u and again applying the same construction.

We then build the sets C, C^+ and C^{++} of all windows of length k of respectively T, T^+ and T^{++}. The set C represents the coding frame while the two others represent shifted frames of the coding frame. Consider the similar sets of windows, F, F^+ and F^{++} associated to another gene. The question is: does the set F which also represents a coding frame exhibit more similarity to the set C than it does to the sets C^+ or C^{++}?

To answer this question we use a simple similarity test based on the radial basis function, which was shown to perform essentially as well as the SVM for this type of problems [17] and which will allow us to derive more information on the structure of the data.

The similarity between two windows X and Y is defined as:

$$S(X,Y) = e^{-\frac{\|X-Y\|^2}{2\sigma^2}}$$

Where σ represents the "tightness" of the similarity measure [17]. The similarity of a vector X to a set V of vectors is defined as the average of the similarities of X to each vector in V.

Given three sets of previously classified training vectors, C which represents a coding frame, C^+ and C^{++}, which represents the shifts of the coding frame, a vector X will be predicted as being a coding vector if it is more similar to C than it is to C^+ or C^{++}. Else it will be predicted as a non-coding vector. When σ decreases, it creates conditions leading to overfitting as two vectors need to be closer in order to have a non-negligible value for the measure of their similarity. In general automatic classification works poorly in case of overfitting, we will see here an interesting example of its use.

5. Comparing Frames

The full results mentioned in that section, as well as the programs used are to be found on the site (http://cs.coastal.edu/ucf/).

We arbitrarily selected the coding sequence of a well curated gene, DKEYP-117E10.6 a gene from the zebrafish. From the T representations of this coding sequence, we derived the three sets of windows C, C^+ and C^{++}. We tested sets of windows F, F^+ and F^{++} derived from the T representations of a few other coding sequences from the same organism. As a starting point we used the representations of the entire coding sequences, and chose the window size as k = 200.

We initially found some confusion where windows from F^+ were seen to be more similar to windows from C, C^+ or C^{++}, nevertheless it seemed that there was a trend, and in particular none of the windows from F was more similar to windows from C^{++}. In order to analyze this further we decided to remove C^+ from the training set.

We then saw something very striking and consistent over the few examples that we ran (see an example in table 1). First when testing F, the set corresponding to the coding frame of the gene, there is a 100% success, we have no false negatives. Furthermore this success rate is maintained up to very small values of sigma, implying that all windows of F are very close to the windows of C, as the results resist the move towards overfitting. On the other hand the results for F^+ and F^{++} varied, and decreased as the value of sigma decreased, indicating more widely distributed vectors.

Sigma	F Similarity	F^+ Similarity	F^{++} Similarity
.4	100%	77.96%	18.79%
.2	100%	77.96%	18.79%
.1	100%	77.96%	18.79%
.01	100%	77.66%	17.01%
.006	100%	73.96%	16.12%
.0058	100%	73.96%	16.12%
.005	100%	72.93%	14.79%
.003	100%	55.33%	7.1%
.002	100%	43.93%	10.06%

Table 1. Percentage of windows from the frames of the fimD gene of yersinia pestis KIM that are similar to windows from the coding frame of the gene DKEYP-117E10.6 from the zebrafish.

This led us to define a first algorithm, that we call the *strict algorithm*,

Strict Algorithm:

We predict that the set F represents the coding frame if and only if

1: for a full range of values of σ all windows of F are more similar to the set C of windows in the coding frame of the training set than they are similar to the windows in the set C^{++} which represents a twice shifted coding frame

2: there exist windows in the twice shifted frame F^{++} that are more similar to the windows in C^{++} than to those in C.

We are simplifying the algorithm by not analyzing the similarity with C^+. The justification, besides being empirical, is based on the following argument: As we require that F be most strongly similar to C, if F^+ is not as similar to C it can be ignored. The case remains where F is also most strongly similar to C. In that case both F and F^+ are most dissimilar to C^{++}, but if we assigned F^+ to the coding frame we would have to assign F to C^{++}, but F exhibits 0% similarity to C^{++}.

We then selected coding sequences from 34 prokaryotes, 12 eukaryotes and 13 archaea. From each of these organisms we selected randomly an average of forty coding sequences. This allowed us to see similarities between coding sequences in the same organism as well as similarities between coding sequences from different domains. We also added 100 genes from KEGG and the Weizmann Institute, which are particularly well-studied and curated coding sequences. Finally we took 953 coding sequences from

a wide variety of mammalian organisms, and with a wide range of GC content, which were previously used as benchmark test sets for gene-finding by the bioinformatics group at the University of Pennsylvania, these three subsets gave us a total of 2939 testing sequences.

The results were striking: 95% of the T-representations of the coding frames of these 2939 coding sequences are more similar to the T representation of the coding sequence of the gene DKEYP-117E10.6 than they are similar to the T representation of its coding sequence shifted twice. Furthermore the strictness of the algorithm requiring no false negative (100% score in the first column) for a range increasingly small values for σ indicates that all these representations are indeed very similar. One factor is that the number of occurrences of codons from T0 is higher in the coding frame, which is consistent with prior results concerning prokaryotes and eukaryotes [5], but as we will see later, it is not the only factor. The failed predictions were found to be mostly concentrated in a few specific organisms, such as Saccharomyces cerevisiae and Caenorhabditis elegans, rather than being randomly distributed, nevertheless the predictions were correct for the vast majority of the other genes in these organisms. We also experimented with other coding sequences for our training set, such as the human TP53 gene, e-coli metE gene and the Pyrococcus abyssi PAB0437 gene and obtained essentially similar results. These training genes are from the three different families and have substantially different DNA sequences.

Now we address the problem of the relevance to comma free codes, codes with bounded parasitism, circular codes, and the notion of testable by fragment. There are a number of striking instances where we find that not only all windows from the coding frame of the tested gene are similar to the windows of the coding frame of the training gene, but none other are. This property would be consistent with the existence of a comma free code made of words of length at most 600 nucleotides, as we have windows of length 200 in the T representations. Or equivalently this would be consistent with the existence of a code made of shorter words, not comma free but having the property of bounded parasitism. This property would limit the possibility of alternative splicing. We also find examples where both F and F^+ show extreme similarity with the windows of the coding frame of the training gene. This is consistent with the eventual possibility of alternative splicing, and corresponds to the notion of spread parasitism: two valid translations are possible. These are important problems that will require our attention, but at present they are beyond the scope of our study, as they leads us to look for special methods to determine the frame related to subfamilies, while we are concerned here with universality.

But in all cases our results support the argument that in the set of coding sequences and shifted coding sequences, the language of coding sequences is testable by fragments. This is because we analyze all windows, and can make decisions solely from this analysis.

We can now address Crick's revised hypothesis: what is the length of the shortest fragment of a coding region that will allow us to predict the frame, independently of the gene or the organism it comes from?

We will have to perform a double fragmentation: first generate a random fragment from a coding sequence, and then fragment again by creating windows as we did previously. But we will change the algorithm, indeed the selection of small fragments implies smaller windows, and this will violate the non false negative requirement: small windows from the coding frame of the tested fragment might be similar to small windows from the twice shifted coding frame of the training set. Furthermore the robustness to overfitting that was displayed previously might not occur as systematically: all scores may vary with decreasing values of σ. Nevertheless it may still be possible to correctly predict the frame, but with less accuracy. So we will use a relaxed form of the algorithm:

Relaxed Algorithm:

We will predict that F is the set of windows extracted from the coding frame if and only if the following conditions are met:

$$1: \quad F_S^{++} < F_S > 50$$

$$2: \quad F_S^{+} - F_S \leq F_S - 50$$

Where F_S, F^+_S, and F^{++}_S are the average scores of respectively F, F^+ and F^{++} for a range of sigma values.

Here instead of requiring that all windows of F be similar to those of C we only require that at least 50% be similar to those of C. Then we require that the windows of F be more similar to those of C than the windows of the twice shifted frame F^{++}. Finally we use a heuristic which is a relaxed version of the preceding one. It is also justified pragmatically, even if its supporting argument is somewhat weaker. The larger F_S is, the less likely F is to be associated with the twice shifted coding frame even if F^+_S is larger than F_S.

Once the size of the fragments to test is chosen, we randomly generate a fragment of that size for each of

the 2939 sequences. The relaxed algorithm allows us to predict the correct frame in 75% of the cases, for a fragment length of ten trinucleotides and a window size of two trinucleotides. The relaxed algorithm also allows us to predict the correct frame in 90% of the cases, for a fragment length of sixty trinucleotides and a window of twenty-five trinucleotides. Due to the randomness of the selection, minor variations in the success rate occur when repeating the process. For these fragments that are substantially smaller than the whole coding sequence, the distribution of the codons from T0 does not necessarily favor as strongly the coding frame. Now even for small window sizes we still see, not as drastically as with windows of size 200, the phenomenon of robustness with respect to overfitting, indicating that windows in the coding frames of most of the genes considered have a very tight relationship.

6. Conclusion

Provided that we replace the notion of comma-free by the related notion of testable by fragment, Crick's 1957 hypothesis seems vindicated: our results support the existence of a universal frame based on a simple mathematical model. Now it is very tempting to try our method on non coding parts of genomes. But one should realize that when we work within the coding region, we know that there exists a coding frame. Outside of the coding region, we will of course find one frame that will be more similar to a coding frame than the two other shifted frames. So one has to adapt our method to a far more complex situation, and this will be a major undertaking. We can nevertheless see indications that it can be useful. For instance preliminary results show that it is sensitive to (obviously) pseudo genes and gene complements, but also seems sensitive to some UTR's.

References

[1] F. H. C. Crick, J. S. Griffith, L. E. Orgel, "Codes Without Commas", *Proc. Natl. Acad. Sci. U.S.A.* 43, 1957, 416-421.

[2] H. Brian, "The invention of the genetic code", *American Scientist* 86, 1998, 8-14.

[3] R. D. Knight, S. J. Freeland, L. F. Landweber, "Selection, History and Chemistry: The Three Faces of the Genetic Code", *Trends Biochem. Sci.* 24, 1999, 241-247.

[4] M. Arita, *Aspects of Molecular Computing,* 2950 Springer Berlin, 2004, 23-35.

[5] D. G. Arquès, C. J. Michel, "A Complementary Circular Code in the Protein Coding Genes", *J. Theor. Biol.* 182, 1996, 45-58.

[6] D. G. Arquès, C. J. Michel, "A Code in the Protein Coding Genes", *BioSystems* 44, 1997, 107-134.

[7] D. G. Arquès, J. Lacan, C. J. Michel, "Identification of protein coding genes in genomes with statistical functions based on the circular code", *BioSystems* 66, 2002, 73-92.

[8] G. Frey, C. J. Michel, "Circular Codes in Archaeal Genomes", *J. Theor. Biol.* 223, 2003, 413-431.

[9] G. Frey, C. J. Michel, "Identification of circular codes in bacterial genomes and their use in a factorization method for retrieving the reading frames of genes", *Comp. Biol. & Chem.* 30, 2006, 87-101.

[10] J-L. Lassez, "On the Structure of Systematic Prefix Codes", *Int. J. Comp. Math.* 3, 1972, 177-188.

[11] J-L. Lassez, "Circular Codes and Synchronization", *Int. J. Comp. & Infor. Sci.* 5, 1976, 201-208.

[12] T. Head, "Splicing Representations of Strictly Locally Testable Languages", *Discrete Appl. Math.* 87, 1998, 139-147.

[13] F. H. C. Crick, S. Brenner, A. Klug, G. Pieczenik, "A Speculation on the Origin of Protein Synthesis", *Origins of Life* 7, 1976, 389-397.

[14] M. Eigen, P. Schuster, *Naturwissenschaften* 65, Springer Berlin, 1978, 341-369.

[15] Google scholar [circular codes]

[16] L. Kari, M. Daley, G. Gloor, R. Siromoney, L. F. Landweber, *Foundations of Software Technology and Theoretical Computer Science* 1738, Springer Berlin, 1999, 269-282.

[17] A. E. Bernal, T. Karadeniz, K. Hospevian, J-L. Lassez, *Advances in Intelligent Data Analysis V* 2810, Springer Berlin, 2003, 187-19.