Statistical algorithms for long DNA sequences: Oligonucleotide distributions and homogeneity maps

P. Katsaloulis[a,b], T. Theoharis[a,*] and A. Provata[b]
[a]*Department of Informatics, University of Athens, 15784 Athens, Greece*
[b]*Institute of Physical Chemistry, National Research Centre "Demokritos", 15310 Athens, Greece*

Abstract. The statistical properties of oligonucleotide appearances within long DNA sequences often reveal useful characteristics of the corresponding DNA areas. Two algorithms to statistically analyze oligonucleotide appearances within long DNA sequences in genome banks are presented. The first algorithm determines statistical indices for arbitrary length oligonucleotides within arbitrary length DNA sequences. The critical exponent μ of the distance distribution between consecutive occurrences of the same oligonucleotide is calculated and its value is shown to characterize the functionality of the oligonucleotide. The second algorithm searches for areas with variable homogeneity, based on the density of oligonucleotides. The two algorithms have been applied to representative eucaryotes (the animal *Mus musculus* and the plant *Arabidopsis thaliana*) and interesting results were obtained, confirmed by biological observations. All programs are open source and publicly available on our web site.

1. Introduction

During the last few years new revolutionary experimental methods in molecular biology have been discovered. It is now possible to sequence DNA macromolecules with increased speed and accuracy. This has resulted in an explosive growth of the amount of biological data being stored in biological databases (such as [6,27]). We now have complete genomic sequences, even for organisms such as human (*Homo sapiens*) and mouse (*Mus musculus*) with extensive genomes.

It is anticipated that, at today's rates, the amount of data inserted into biological databases will double every 18 months. It is clear that this tremendous amount of data is of no value, unless there exist tools for effectively searching and manipulating it. For this reason various biological packages have been developed, such as BLAST [3,37], FASTA [23,33], CLUSTAL [12,34], while other numerical approaches and algorithms are presented in [1,2,4,5,8–10,13–15,18,24,26,28].

Most of the problems addressed by these packages deal with finding specific patterns in DNA or protein sequences, searching for similarities between known sequences, querying biological databases for similar sequences given an unknown one, developing algorithms which try to reconstruct the 3D structure of a given macromolecule, producing tools to automatically distinguish functional areas (like coding/non-coding regions in DNA or topology prediction for proteins), calculating various statistical and mathematical parameters etc. Despite the increasing number of available tools, the problem of categorizing oligonucleotides based on their statistical properties is still open. We propose two algorithms which deal with small DNA sequences and their distribution across the whole chromosome, in order to be able to categorize these sequences or DNA areas only from their statistical properties and not by laboratory biological findings.

Let us first formalize the representation of the DNA sequences used in this work. Genomic DNA sequences consist of the four nucleic acids; Adenine, Cytosine, Guanine and Thymine:

$$Base = \{A, C, G, T\} \tag{1}$$

[*] Corresponding author. Tel.: +30 210 7275106; E-mail: theotheo@di.uoa.gr.

ISSN 1058-9244/05/$17.00

although in other genomic macromolecules other bases are found as well (e.g. Uracil – U in RNA). Inside the biological DNA databases only the above four bases are stored. Thus, a DNA Sequence of length n can be described as:

$$Sequence(n) = Base^n \tag{2}$$

Inside the cell, there is usually more than one long DNA sequence. Each one of these sequences is called a Chromosome, it is independent of the others and usually includes different genomic information. [1]

Again, a chromosome can be represented as

$$Chromosome(c) = Base^c \tag{3}$$

where c would be in the range of a couple of thousand (for simpler organisms) to hundreds of millions (for more complex organisms). If the size of the DNA sequence is small, then this is called *Oligonucleotide*:

$$Oligonucleotide(m) = Sequence(m), \quad \text{where} \quad m \ll c \tag{4}$$

Although the main operation of the DNA is to code for proteins, it is known that, in higher eucaryotic organisms, only a small percentage of the DNA is translated, in order to produce proteins. These areas are called *coding areas*. The rest of the DNA has more structural than functional role and such areas are called *non-coding areas*. In order for the coding areas to be distinguishable by the enzyme which promotes the transcription,[2] usually in the beginning of the coding regions there is a special DNA sequence called the *promoter*. Promoters usually have a length in the order of hundreds of bases. Between two successive appearances of the promoter there are at least one coding and one non coding sequence. Inside each promoter there are small oligonucleotides of length $m = 2 \dots 10$, which are steadily present, called *consensus sequences*. Known consensus sequences in eucaryotes are, among others, the `CG` and the `TATA` sequences.

In a recent work, the distance distribution between two consecutive appearances of a given oligonucleotide has been calculated [16]. The investigation has been performed on human chromosomes 21 and 22 for oligonucleotides of length $m = 5$ and $m = 6$. It has been found that the oligonucleotides that contain consensus sequences of promoters follow long tailed distributions

$$P(S) \sim S^{-1-\mu}, \qquad 0 \leqslant \mu \leqslant 2 \tag{5}$$

where S is the distance between two consecutive occurrences of the same oligonucleotide sequence, $P(S)$ is the distribution of S and μ is called the power law or critical exponent. In contrast, randomly generated oligonucleotides follow short tailed distributions.

Encouraged by this finding, we decided to generalize the process; we created two algorithms, the *Oligonucleotide Process Algorithm* (OPA) and the *Statistical Homogeneity Map* (SHMap) algorithm. The former one (OPA) calculates statistical indices of oligonucleotide distributions in long DNA sequences. Representative indices are the frequency of appearance, the maximum distance, the average distance, the distance deviation between two occurrences of the same oligonucleotide and the power law exponent μ. The values of these indices are shown to be associated with the degree of functionality of the corresponding oligonucleotide indicating whether the particular oligonucleotide sequence serves as promoter signature for this organism. The latter (SHMap) algorithm maps areas in the DNA sequence which lack homogeneity, providing information about the characteristics of the underlying DNA sequence and possibly predicting its functionality.

Each one of these algorithms is applied to chromosomal data (DNA sequences with c at least 10^3) and statistically manipulates these long sequences. A general interface has been developed which is able to input a DNA sequence in either plain text format, without any special coding, or in NCBI's FASTA (FNA) format [23]. All the code that resulted from this work are publicly available under an open source licence (GNU GPL) through our web site [32].

In the next two sections we describe the OPA and the SHMap algorithms respectively. In Section 4 we discuss the applications of the two algorithms and the external tools required. In Section 5 we present interesting results from the application of our algorithms to real biological data. Finally in Section 6 we discuss the results, address some open problems and propose future extensions.

2. Oligonucleotide Processing Algorithm (OPA)

Our aim here was to examine the statistical properties of oligonucleotides within a given chromosome

[1] Sometimes during the life of the cell (e.g. mitosis metaphase), when the cell is going to double, each one of these sequences doubles and there are two appearances of the same sequence.

[2] Like RNA polymerase II.

(sequence). To this end our algorithm aims to distinguish oligonucleotides with special statistical properties. It is then extremely useful to compare these results with experimentally determined oligonucleotides with distinct biological function.

The main points of focus are the following:

- Distance Distribution of two consecutive appearances of the same Oligonucleotide Sequence DDOS inside a give chromosome
- Determination of the power law exponent μ for each oligonucleotide
- Ordering of the results on any statistical parameter calculated above and checking whether the ranking of oligonucleotides has functional meaning or if there is any type of possible clustering between the oligonucleotides

In order to calculate the statistical properties of the oligonucleotides, pattern matching in DNA sequences is needed. Apart from the four symbols which represent the four nucleic acid bases (A, T, C and G), more symbols exist in DNA databases, which are used to describe bases whose composition is not fully sequenced. In the following algorithms we have used a generic approach, where all these symbols are taken into account [20–22]. Table 1 gives the truth matrix we constructed to decide whether two bases match.

To calculate the power law exponent μ we have followed two approaches, depending on the method being used. The first approach relies on the observation that the DDOS for almost every oligonucleotide has a central linear part in double logarithmic scale, usually found between fixed boundaries for a given chromosome. Using this information it is possible to distinguish oligonucleotides depending on their DDOS and their critical exponent μ, which is the slope of this linear part (Eq. (5)). The algorithm variant used is the following:

```
DNA = load dna('DNASequence.FNA');
/* Provide the length M of the oligonucleotides */
M = input();
/* Create a list with all possible oligonucleotides of length M */
/* The list should have $4^M$ items */
LIST = compute possible
 oligonucleotides (M);
/* Go through every item in the list */
for (S in LIST) {
  /* Search for the first occurrence of S in the DNA */
  /* sequence (from position 0) */
  POSITION1 = find(0, S, DNA);
  /* Initialize distances list */
  DISTANCES = new list();
  while (not end of 'DNA') {
    POSITION2 = find (POSITION1+M, S, DNA);
    add distance to list(DISTANCES,
POSITION2-POSITION1);
    POSITION1 = POSITION2;
  }
  /* Calculate the various statistical parameters */
  MAXDIST = calculate max distance
(DISTANCES);
  AVRDIST = calculate average
 distance(DISTANCES);
  DISTDEV = calculate distance
 deviation(DISTANCES);
  HISTOGRAM1 = calculate histogram
(DISTANCES);
  HISTOGRAM2 = calculate cumulative
 histogram (DISTANCES);
  HISTOGRAM3 = convert to loglog
(HISTOGRAM2);
  /* Perform a line fitting over the HISTOGRAM, -1-m is the slope */
  SLOPE = line fitting(HISTOGRAM3);
  /* VAR is any statistical variable, such as MAXDIST, SLOPE, etc. */
  sort (HISTOGRAM3, VAR);
  save (HISTOGRAM3);
}
```

The complexity of this algorithm is $O(4^m * c * m)$, where m is the length of the search pattern and c is the length of the DNA sequence.

The least squares algorithm [25] has been used for the line fitting. The results are ordered by the desired statistical parameter, and the oligonucleotides which appear to have extreme values are exposed. For example, if the critical exponent μ is used for sorting, oligonucleotides with small values of $|\mu|$ (follow long range distributions) are expected to include consensus sequences, whereas oligonucleotides with large values of $|\mu|$, (follow short range distributions) have no clear biological meaning. Interestingly, this ordering is in general robust, for all statistical properties considered.

The second approach uses a general curve in order to fit the produced DDOS, taking into account the whole histogram. We have selected a curve with an exponential and a polynomial part, in order to be able to describe both power law (long tails) and exponential (short tail) behavior of the distribution:

$$y(x) = Ax^{-1-j}e^{-kx} \tag{6}$$

This expression contains three independent parameters: A being a normalization parameter, j being an intermediate scale and k being a large scale parameter. The parameter j corresponds to the critical exponent μ presented above. The curve fitting algorithm which was used is a combined Levenberg-Marquardt [17,19] with Gauss-Newton method. The curve fitting version of the algorithm is:

```
/* Load the DNA into memory, input the length of the */
/* oligonucleotides and check every oligonucleotide in order */
DNA = load dna('DNASequence.FNA');
M = input();
LIST = compute possible
 oligonucleotides (M);
for (S in LIST) {
  POSITION1 = find(0, S, 'DNA');
  DISTANCES = new list();
```

Table 1

	A	B	C	D	G	H	K	M	N	R	S	T	V	W	Y
A	1	0	0	1	0	1	0	1	1	1	0	0	1	1	0
B	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0
C	0	1	1	0	0	1	0	1	1	0	1	0	1	0	1
D	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0
G	0	1	0	1	1	0	1	0	1	1	1	0	1	0	0
H	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0
K	0	1	0	1	0	0	1	0	1	0	0	0	0	0	0
M	0	0	0	0	0	1	0	1	1	0	0	0	1	0	0
N	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R	0	0	0	1	0	0	0	0	1	1	0	0	1	0	0
S	0	1	0	0	0	0	0	0	1	0	1	0	1	0	0
T	0	1	0	1	0	1	1	0	1	0	0	1	0	1	1
V	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0
W	0	0	0	1	0	1	0	0	1	0	0	0	0	1	0
Y	0	1	0	0	0	1	0	0	1	0	0	0	0	0	1

The DNA base matching matrix of two nucleic acid symbols. The symbols in the vertical axis represent the bases found inside the DNA sequence being searched and the symbols in the horizontal axis represent the bases found in the search pattern ('1' = match, '0' = no match). We note that most of these symbols match with more than one symbol. The generic symbols match the more specific ones only when the former appear in the search pattern, not inside the long DNA sequence. The representation of each symbol is as follows: G for Guanine, A for Adenine, T for Thymine, C for Cytosine, R for G or A (puRine), Y for T or C (pYrimidine), M for A or C (aMino), K for G or T (Keto), S for G or C, W for A or T, H for A or C or T, B for G or T or C, V for G or C or A, D for G or A or T, and N for G or A or T or C [20–22].

```
   while (not end of DNA) {
      POSITION2 = find (POSITION1+M, S, DNA);
      add distance to list(DISTANCES,
POSITION2-POSITION1);
      POSITION1 = POSITION2;
   }
   /* Calculate the distance distribution of this oligonucleotide */
   HISTOGRAM = calculate histogram
(DISTANCES);
   /* Convert the histogram in double logarithmic scale */
   convert to  loglog (HISTOGRAM);
   save (HISTOGRAM);
}
/* Parse all histograms being produced */
for (H in HISTOGRAM files) {
   /* Perform the curve fitting on this histogram */
   curve fit (H);
   /* Store the curve parameters */
   store (A, j, k);
}
/* Perform a data clustering based on j,k */
find clustering(j,k);
display data();
```

The complexity of this algorithm is again $O(4^m * c * m)$ as in the previous version of the algorithm.

This approach presents new views over the possible classification of the oligonucleotides. By appropriately mapping the various parameters, clustering of oligonucleotides may appear, when statistically meaningful queries are posed. In order to evaluate the produced results, we have taken into account only the j and k parameters, since these describe the statistical behavior of the result. As can be seen in Section 5.2, two clearly distinct areas are visible and this observation is robust in both chromosomes we have chosen.

We would like to note that the sequences which are found with this algorithm do not appear solely inside the promoters, but can also be seen elsewhere in the DNA, like inside exons, and thus do not appear exclusively at the beginning of genes. Since this algorithm calculates statistical indices which refer mostly to the tails of the size distribution of these sequences, at least the consensus sequences of the promoters which have large interdistances (i.e. separate mostly intergenic regions) correspond to promoter sequences.

3. Statistical Homogeneity Map (SHMap)

The algorithms presented above give statistical information on various oligonucleotide combinations. It is also useful to be able to map areas inside the chromosome according to their statistical behavior. We thus consider another biological observation; the *lack of homogeneity* within eucaryotic chromosomes. Each eucaryotic chromosome consists of areas with different composition. Some areas can be described as "random" from the statistical point of view, whereas other areas have more "stable" consistency [26].

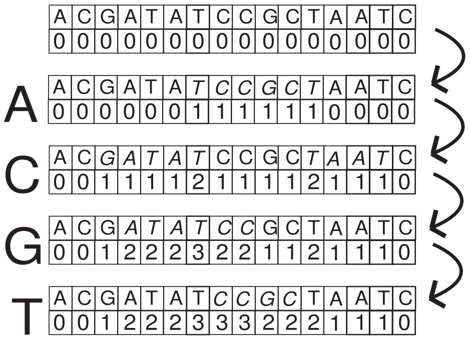
The algorithm proposed here marks areas of the chromosome according to their "randomness". As a base measure we employ all possible oligonucleotides of length m. We distinguish the areas which are rich in different oligonucleotides, and those which consist of only a few oligonucleotides. The algorithm is as follows:

```
/* Load the DNA into memory, input the length of the */
/* oligonucleotides and check every oligonucleotide in order */
DNA = load dna('DNASequence.FNA');
/* Allocate memory in order to save the SHMap values */
MAP = allocate memory(size of(DNA);
clear memory (MAP);
M = input();
LIST = compute possible
 oligonucleotides (M);
for (S in LIST) {
   POSITION1 = find(0, S, DNA);
   while (not end of 'DNA') {
      POSITION2 = find (POSITION1+M, S, DNA);
      DISTANCE = POSITION2-POSITION1;
      /* Check if the distance between two consecutive appearances */
      /* of an oligonucleotide are above a given threshold */
      if (DISTANCE > THRESHOLD) {
         /* Mark all positions between first and second */
         /* appearance of the oligonucleotide */
         for (K between POSITION1 and
POSITION2) {
            MAP[K] = MAP[K]+1;
         }
      }
      /* The second position becomes now the first */
      POSITION1 = POSITION2;
   }
}
/* Normalize the MAP between the values 0 to 255 */
normalize map (MAP, 0, 255)
save map(MAP);
```

The result of this algorithm is a data file, with the same size as the input DNA sequence; there is a one-to-one correspondence between chromosome bases and values within this file (Fig. 1). The complexity of this algorithm is $O(4^m * c * m)$, since, for each one of the 4^m oligonucleotides, the start of each new search is the end of the previous one (where the oligonucleotide was last found). In the end the entire genome c is traversed once.

The biological meaning of values within this file is as follows:

- a lower value implies that the distances between oligonucleotides in this area are generally smaller than the given threshold. Having short distances means that the possibility of finding any given combination, starting from any position inside this area is high, or in other words, that most combinations are present and mixed in this area. Since this kind of behavior resembles "random" distribution, it is also expected that these areas include mostly coding DNA sequences.
- a higher value implies long distances between oligonucleotides (above the given threshold). Having long distances means that it is less probable to find the next occurrence of a certain oligonucleotide inside this area. Since we do not consider extensive DNA gaps in this implementation of the algorithm, but there is a contiguous coverage of bases, the reason for the long distances is the over-representation of few specific oligonucleotide combinations in this area, forcing the remaining majority of the oligonucleotides to be under-represented. This behavior is common in non-coding DNA sequences, where the presence of structures like poly-A (long sequences consisting only of adenine) are common.



	A	C	G	A	T	A	T	C	C	G	C	T	A	A	T	C
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A	A	C	G	A	T	A	*T*	*C*	*C*	*G*	*C*	*T*	A	A	T	C
	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
C	A	C	*G*	*A*	*T*	*A*	*T*	C	C	G	C	*T*	*A*	*A*	*T*	C
	0	0	1	1	1	1	2	1	1	1	1	2	1	1	1	0
G	A	C	G	*A*	*T*	*A*	*T*	*C*	*C*	G	C	T	A	A	T	C
	0	0	1	2	2	2	3	2	2	1	1	2	1	1	1	0
T	A	C	G	A	T	A	T	*C*	*C*	*G*	*C*	T	A	A	T	C
	0	0	1	2	2	2	3	3	3	2	2	2	1	1	1	0

Fig. 1. Schema of the SHMap algorithm. Top lines contain the DNA sequence and bottom lines contain the MAP values. In this example we consider for simplicity single-base nucleotides (mononucleotides, $m = 1$) and thus the number of possible oligonucleotides are $4^m = 4^1 = 4$. The four sequences are A, C, G and T respectively. The threshold in this example is $4^{m+d} = 4$, taken $m = 1$ and $d = 0$ (Eq. (7)). Each new row in this schema depicts the status of the MAP after a step of the algorithm for the respective oligonucleotide (shown on the left of the table). The bases in *italics* mark the positions in the DNA sequence which will be incremented.

The choice of the threshold is important, as it distinguishes whether the distance between two occurrences of an oligonucleotide is statistically insignificant or not. As c is the length of the DNA sequence and m is the size of the oligonucleotides, then the number of m-sized oligonucleotides inside c are $c - m + 1$. Since the number of possible oligonucleotides is 4^m, the expected number of appearances of each oligonucleotide within a random DNA sequence of size c is $\frac{c-m+1}{4^m}$. The average distance between two consecutive appearances of a specific oligonucleotide is expected to be

$$Distance(c, m) = \frac{c - m + 1}{\frac{c-m+1}{4^m}} = 4^m \qquad (7)$$

In this study the threshold was set to 4^{m+d}, where d is used to bring the threshold well above the random probability of appearance (4^m).

Using this algorithm it is possible to distinguish areas which are rich in oligonucleotide combinations (lower value) from those which are poorer (higher values).

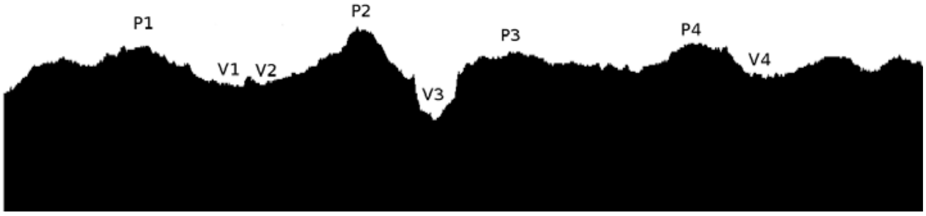


Fig. 2. Application of the SHMap algorithm over chromosome 1 of *Mus musculus* in the `NT 039170` area. The size of the oligonucleotide being used was $m = 5$. The region between 11730700–11813700 is displayed. Some of the peaks appear to be in the regions 11740900–11742700 (P1), 11760000–11763500 (P2), 11775800–11776800 (P3), 11793000–11794100 (P4) and some of the lowest valleys appear to be in the regions 11748600–11750200 (V1), 11753400–11755000 (V2), 11765500–11766700 (V3) and 11798000–11800000 (V4). By direct comparison with the NCBI gene database it is found that in general the peaks correspond to the introns, while the valleys correspond to regions rich in exons of the Dst gene. The corresponding regions (which include the identified peaks) are (11740853–11742704) for P1, (11754490–11763711) for P2, (11775756–11776896) for P3 and (11792763–11794229) for P4. The valleys correspond to several exons each, with statistically insignificant small introns between them: [11748859–11749016], [11749445–11749623] and [11750055–11750193] for V1, [11753810–11753917] and [11754392–11754490] for V2, [11765657–11765745] and [11766514–11766646] for V3, [11798030–11798271] and [11799091–11799220] for V4.

Since the coding regions appear to be richer in oligonucleotide combinations, we expect them to be inside the areas with the lower values. A visualized result of this algorithm is presented in Section 5.3 (Fig. 2). Since this approach is statistical and not biochemical, there is less accuracy in the positioning of the exons and introns. It can be used as a tool to point to DNA areas which need to be further investigated by traditional biochemical methods.

4. Implementation

An integrated application under an open source licence (GNU GPL) implements the above algorithms. It is console-based, although some components (such as the display of various plots) produce graphical output. Our development environment was a Linux single processor (Intel) system. We used the ANSI C++ language under the GNU G++ compiler in order for the source code to be portable [29]. We have also tested it under Windows 98 and Windows XP environments using cygwin/mingw32 tools. Other tools used include the BASH shell [35] to manage and sort the results, the GNUPLOT utility [36] to graphically present the results and the GRACE application [31] to perform the curve fitting over the data.

Depending on the amount of computation and the plan of work, it can be used in either interactive or batch mode.

4.1. Interactive

This mode is the default. The user is able to interactively perform various exploratory statistical tests in real time. Typically a single oligonucleotide is tested at a time. The input DNA sequence is either in plain text or in FNA format. A search pattern is specified and statistical information is displayed in real time, such as frequency of appearance, maximum and average distance, distance deviation and the critical exponent μ. It is also possible to calculate simple or cumulative distributions and to graphically display a plot of distances for the given oligonucleotide combination or the DDOS together with the calculated slope.

4.2. Batch

This mode is useful for collecting statistical data on multiple oligonucleotides. The algorithms described in the previous sections are implemented in batch mode. Both the OPA (with line or curve fitting) and the SHMap algorithm can be executed on data provided by the user at run time. For the visual display of the clustering of oligonucleotides, the sequences can be split according to a regular expression (RegExp [7]), as shown in Figs 3 and 4. The computational cost of each algorithm on a Pentium 4 PC (2.5 GHz) for quintuplet processing on Chromosome 19 of *Mus musculus* (about 60 Mbases) was of the order of 1 hour.

In addition to the presented algorithms, a few extra facilities are also available. It is possible to calculate all histograms for an oligonucleotide string of a specified length and store the results in a *single* file. The scales of the histograms are not normalized (top part of Fig. 5). A 2D matrix is created whose horizontal dimension contains the size distribution values and the vertical dimension indexes the oligonucleotide. It is also possible to calculate all histograms in normalized scales (bottom part of Fig. 5). This is useful for comparing the shape of the histograms for different oligonucleotides.

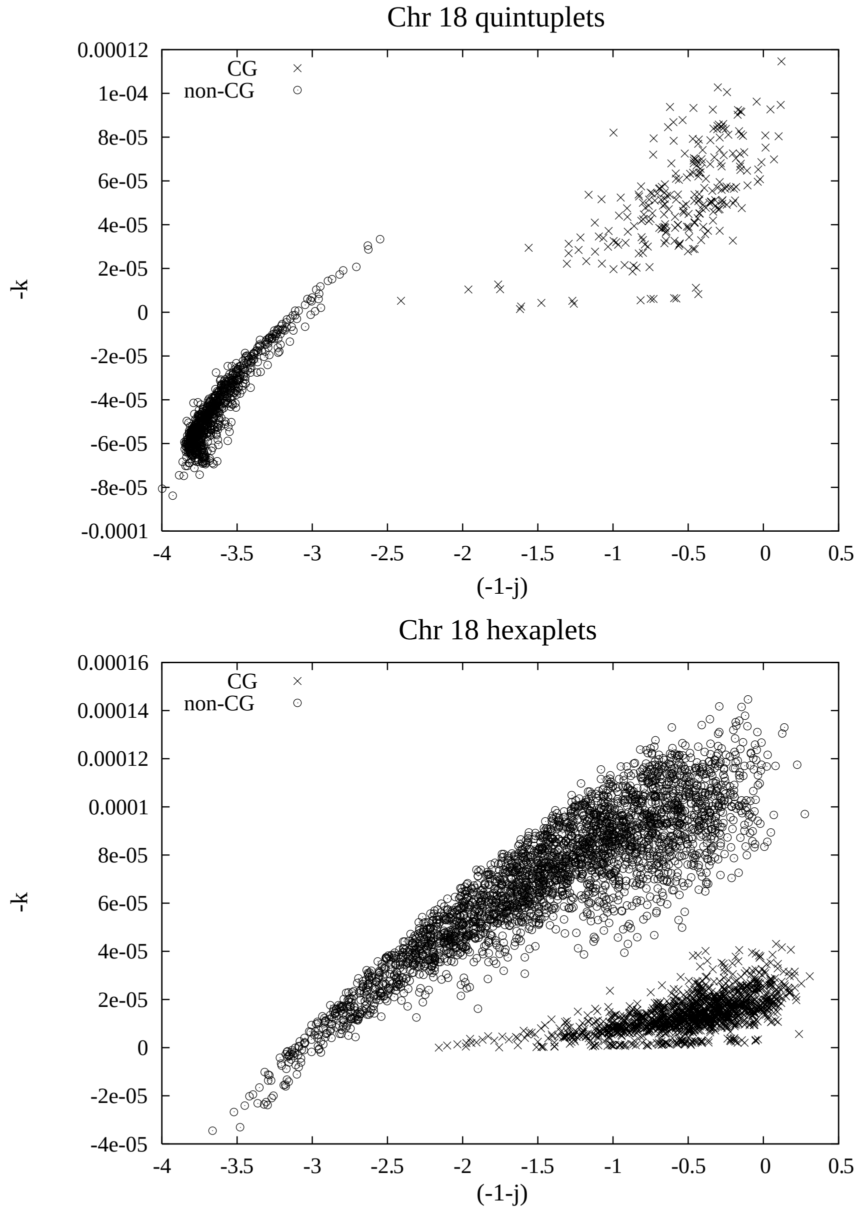


Fig. 3. Application of the OPA algorithm over chromosome 18 of *Mus musculus*. The axes plot $(-1 - j)$ against $-k$. Two clearly separated areas are visible; on one side are oligonucleotides with at least one occurrence of the CG sequence (X) and on the other all other oligonucleotides (O).

5. Application to biological data

To test the usability and effectiveness of the proposed algorithms, we performed tests on biological data. Fully sequenced chromosomes from the NCBI genome bank were obtained [30]. The organisms which were used are the animal *Mus musculus* (mouse) and the plant *Arabidopsis thaliana*.

5.1. Long range distribution of oligonucleotides

In this study, reference chromosomes 1, 15 and 19 of *Mus musculus* and chromosomes 1, 2 and 3 of *Arabidopsis thaliana* from the NCBI database were tested. The line fitting variation of the OPA algorithm was employed to calculate the critical exponent. The linear regions taken into account were in the range 2–6 of the $P(S)$ in double logarithmic scale. Quadruplets

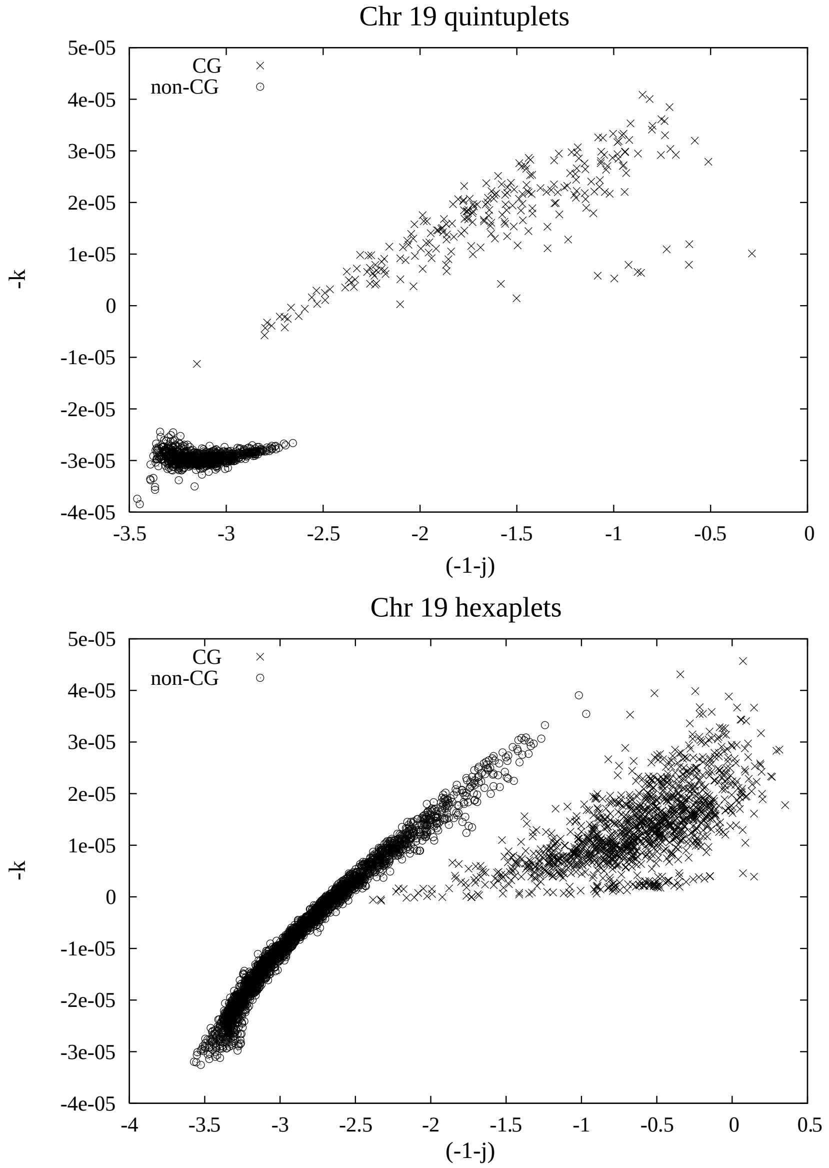


Fig. 4. Application of the OPA algorithm over chromosome 19 of *Mus musculus*. As in Fig. 3, two areas are again visible, on the right are oligonucleotides with at least one occurrence of the CG pattern (X) and on the left all other oligonucleotides (O).

(oligonucleotides of $m = 4$) and quintuplets (oligonucleotides of $m = 5$) were considered. The oligonucleotides were sorted according to the value of their critical exponent μ and the combinations with the lowest and the highest values of μ are presented here. An example of the produced output can be seen in Table 2.

In all tested chromosomes of *Mus musculus* it can be seen that in general the quintuplets with the smallest absolute value of μ are those which contain the sequence CG twice. Various combinations of this basic pattern appear to belong to this group, such as TCGCG and CGCGA (which are complementary), CGCGT and ACGCG (complementary), CGTCG and CGACG (complementary) and a few others. Following the same pattern, the quadruplet with the smallest absolute value of μ is the one with the double CG sequence, namely the CGCG. The oligonucleotides with the highest value of $|\mu|$ do not appear to have any specific pattern. We note here that the complex CG is a consensus sequence of the RNA polymerase II promoter in some organisms. The OPA algorithm, without any input about promoter structure, solely based on distance distribution between

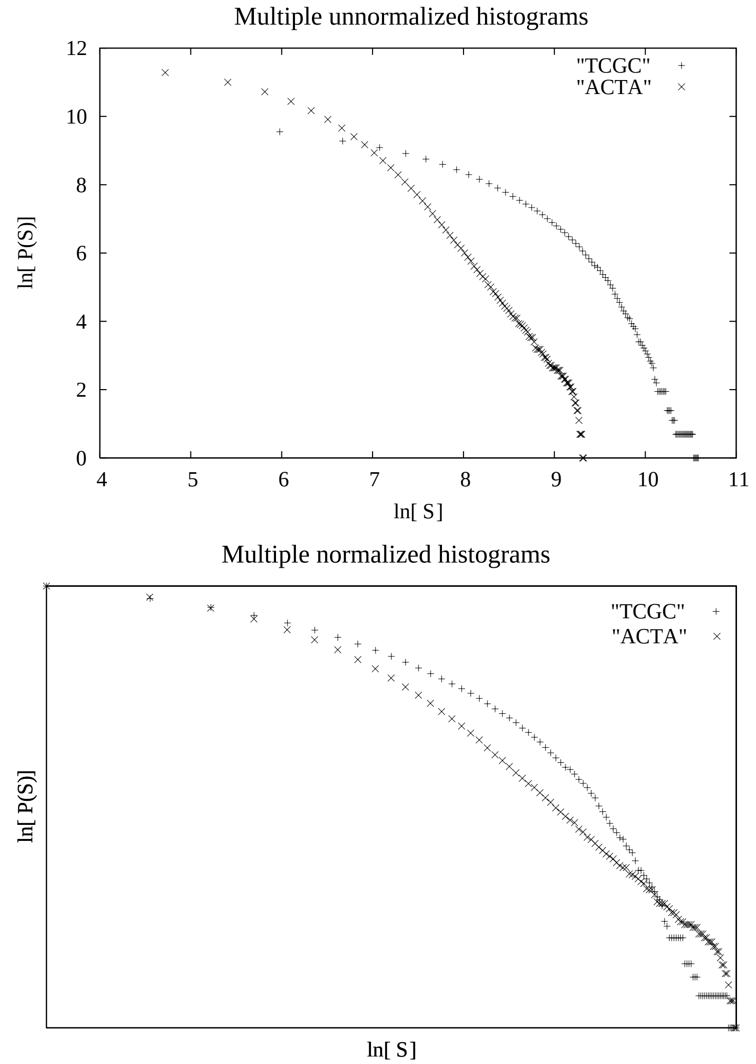


Fig. 5. Calculation of multiple histograms. The top plot displays the unnormalized and the bottom the normalized histograms. For clarity of presentation only two of the 256 histograms are shown (the actual number of histograms calculated are 4^n, where $m = 4$ in the case of quadruplets). The displayed oligonucleotides are ACTA and TCGC from the 10th chromosome of *Mus musculus*.

various oligonucleotides has sorted out all the oligonucleotides which contain the signature of the promoter of the polymerase.

In *Arabidopsis th.* the situation is more complicated. In chromosome 1 the quadruplet with the smallest value of $|\mu|$ is the TATA, which is different to the one found for *Mus m.* In quintuplets we have a similar situation. The sequences with the smallest value of $|\mu|$ are those which have the TATA sequence or point mutations of it. Sequences with large values of $|\mu|$ do not appear to follow any special pattern, although some of the oligonucleotides appear to have A and T, but with different ordering than the one described above.

In chromosomes 2 and 3 the quadruplets with the smallest value of $|\mu|$ are the CGCG and GCGC sequences. The results are similar for quintuplets; sequences with the smallest value of $|\mu|$ appear to contain the CG sequence twice. These sequences are CGCGC, GCGCG and point mutations of them, rich in cytosine and guanine. Although at first sight the results appear to contradict (since in both organisms it is the same enzyme which promotes the production of mRNA) they can easily be explained in biological terms. It has been found that in mammals and other higher organisms, the consensus sequence of the promoter includes the CG sequence. However in *Arabidopsis th.* the TATA

Table 2

% SEQ	FREQ	AVG	MAX	STDDEV	FIT	CFIT	LNFIT
CTCA	17856	239	2503	252.545	0.0388571	−0.00792694	−5.85913
AGAG	22392	190	2724	225.815	−0.0166103	−0.0120101	−5.81601
TCAG	17476	244	2856	255.668	−0.0175687	−0.00545257	−5.79798
AGGT	12641	338	4149	351.576	0.0434072	−0.00367821	−5.65297
CTGA	17755	240	2404	248.927	−0.00226716	−0.00842822	−5.58917
AAGC	11951	357	3730	387.117	0.0109258	−0.0405405	−5.56692
ACTT	14646	291	3229	306.616	−0.00621426	−0.0154523	−5.56439
GTGA	12892	331	4088	343.58	−0.0101775	−0.00443319	−5.52546
AGGA	19309	221	2576	248.277	−0.00600354	−0.00443272	−5.52412
TGAG	18476	231	2325	248.305	0.02659	−0.0109548	−5.50157
...	...	...	...	...	...	...	...
TCGG	1757	2433	28958	3072.47	−0.000182006	−0.00024667	−2.30993
GGCG	1976	2164	38312	3337.87	−0.00177118	−0.00043499	−2.30579
CACA	22612	189	5063	257.518	−0.0078082	−0.00230276	−2.25369
CGGA	1940	2203	28078	2976.34	0.000635356	−0.000635895	−2.22556
CGGC	1836	2329	34029	3510.3	−0.000213448	−0.000323557	−2.21539
ACGC	1648	2593	27919	3340.61	−0.00113418	−0.00101467	−2.19301
CGCC	1971	2168	41934	3379.38	8.54279e-05	−0.000357682	−2.05837
CCGC	1642	2602	45587	4017.48	−3.65871e-05	−0.00047312	−2.01543
GCGC	1449	2950	57166	4963.49	−0.000699545	−0.000175142	−2.01383
CGCG	533	7998	108882	14786.1	−0.000275482	−0.000127624	−1.16537

Sample output of the OPA algorithm for *Mus musculus chromosome* 18. The columns are, from left to right, the current oligonucleotide sequence (SEQ), the frequency of appearance (FREQ), the average distance (AVG), the maximum distance (MAX), the deviation of the distances (STDDEV), the slope of the histogram (FIT), the slope of the cumulative histogram (CFIT) and the slope of the cumulative histogram in double logarithmic scale (LNFIT) which is also the value of the critical exponent ($-1-\mu$). The data is sorted according to the critical exponent values. Only the extreme parts of the list are shown. The oligonucleotide with the smallest value of $|\mu|$ is the CGCG sequence.

oligonucleotide is an important consensus sequence, in addition to CG. Our algorithms confirm this biological particularity.

From our current and previous studies we have seen that the sequences which follow long-range distributions were found to correspond to consensus sequences of DNA promoters. Thus at this stage the OPA algorithm might be used to predict possible consensus promoter sequences in long DNA sequences.

5.2. Clustering of oligonucleotides

In this test we perform curve fitting over the DDOS. We have used the SHMap algorithm over chromosomes 18 and 19 of *Mus musculus* and taken into account quintuplets and hexaplets ($m = 5$ and $m = 6$ respectively). The results can be seen in Figs 3 and 4. Each scatter plot depicts the two main parameters $-1 - j$ and $-k$. Every plotted symbol corresponds to a single oligonucleotide. We have divided the 4^m oligonucleotides in two sets, the first set α (marker X) consists exclusively of oligonucleotides which include the CG sequence, and the second set β (marker O) contains all other oligonucleotides.

The result is rather amazing. Two clearly separated clusters are visible in every plot, one consisting solely of oligonucleotides belonging to set α on the right part of the graph, and the other consisting solely of oligonucleotides belonging to set β, on the left part. The existence of the CG sequence is the characteristic differentiator of these clusters; CG is known to be a consensus sequence of the promoter for this organism. This clustering is more prominent in mammals and is less evident in plants and lower eucaryotes.

5.3. SHMap

We have used SHMap to calculate the statistical homogeneity map of chromosome 1 of *Mus musculus*. An example output of the algorithm can be seen in Fig. 2 using oligonucleotides of size $m = 5$. In this example we have focused on the NT 039170 area of this chromosome. Since the whole data sequence is of the order of 10^7, only a specific region is shown in the example (namely bases 11730700–11813700). The threshold being used is $4^{m+d} = 4^{5+2} = 16384$. This area corresponds to the Dst gene, which produces the dystonin protein.

The diagram shows some peaks and some valleys. The peaks characterize low diversity for the underlying DNA sequence and in this example correspond to non-translated areas. The deeper valleys are found to correspond to areas of the gene rich in exons, which means that they are part of the coding regions of this DNA sequence. The algorithm thus has predictive power in non-annotated sequences. It would be interesting to further investigate this behavior and compare it against biological data regarding the functionality of the given chromosomal areas.

At this point we should note that in chromosomes which are not fully sequenced, an artifact may appear. If the DNA sequences have areas of unknown base consistency (e.g. having the symbol N), the unknown bases will not match with any given sequence and will produce false high peaks.

6. Conclusions and open problems

We present a set of algorithms for the statistical analysis of DNA data. Interestingly the use of our algorithms over laboratory biological data revealed a specific behavior related to functionality. Although the approach was completely based on mathematical terms, the sequences which stand out are those which have specific biological meaning (e.g. consensus sequences of promoters). This is important, not only because it is a different kind of proof for the special function of these sequences, but also because they could be used on organisms for which we have the DNA sequence but do not know much about the functionality of their genome.

We would like to note that in this analysis we do not presume the existence of promoters. We have focused only on calculating statistics of long DNA sequences and estimating the size distribution between oligonucleotides. It is true that there can be more than one promoter consensus sequence even for the same RNA polymerase. This situation is depicted in *Arabidopsis th.* where both TATA and CG sequences appear. However since the treatment is statistical, only the prominent promoter consensus sequences dominate. Sequences which appear sporadically do not contribute significantly to statistics and thus may not appear in the top places of the results.

We have to stress that apart from the requirement of long DNA sequences, our algorithms are neither organism nor data specific. They can equally be applied to eucaryotes or procaryotes, ‘higher’ or ‘lower’ organisms. It is expected that the results will vary according to the selection of organism, since each organism might have different enzymes and biological pathways. The main principles of the algorithms will remain, only the biological interpretation of the data will change.

OPA algorithm seems to distinguish the promoter consensus sequences from other oligonucleotides, since these sequences appear to have the smallest absolute value of μ. It may be possible to use this algorithm in order to predict possible promoter consensus sequences in unknown long DNA sequences. In the current work we have focused on making statistical tools which can be used to analyze any sequence in the statistical sense of searching for oligonucleotides.

Although it would be possible to modify these algorithms in order to ignore consecutive repeats of oligonucleotides, the statistics will change. The repeats are an important element in the structure of intergenic regions and in non-coding DNA sequences and they have been produced by evolutionary forces. For this reason they drastically contribute to the statistics of the tails of the sequences and they induce long range properties.

Finally it would be interesting to extend our system so as to be able to zoom in various levels on the map and display specific DNA ranges. It is also important to tag these DNA areas with information such as which known genes are inside this area or which parts consist mainly of exons or introns. An implementation under the GRID [11] would allow the separation of logically distinct parts of our system (data banks, processing, display) and speed up the batch mode through distributed processing.

The algorithms developed can be downloaded and tested from our website [32] under an open source licence (GNU GPL).

7. Acknowledgements

The authors acknowledge financial support from the “HERAKLEITOS” research fellowships program 70/3/7156 with emphasis on basic research, Hellenic Ministry of Education.

References

[1] Y. Almirantis, A standard deviation based quantification differentiates coding from non-coding DNA sequences and gives insight to their evolutionary history, *Theor. Biol.* **196** (1999), 217.

[2] Y. Almirantis and A. Provata, The ‘clustered structure’ of the purines/pyrimidines distribution in DNA distinguishes systematically between coding and non-coding sequences, *Bull. Math. Biol.* **59** (1997), 975.

[3] S.F. Altschul, W. Gish, W. Miller, E.W. Myers and D.J. Lipman. Basic local alignment search tool, *Mol Biol* **215** (1990), 403.

[4] A. Arneodo, E. Bacry, P.V. Graves and J.F. Muzy, Characterizing long-range correlations in DNA sequences from wavelet analysis, *Phys. Rev. Lett.* **74** (1995), 3293.

[5] A. Arneodo, Y. d'Aubenton Carafa, E. Bacry, P.V. Graves, J.F. Muzy and C. Thermes, Wavelet based fractal analysis of DNA sequences, *Physica* **D96** (1996), 291.

[6] D.A. Benson, M.S. Boguski, D.J. Lipman, J. Ostell, B.F. Ouellette, B.A. Rapp and D.L. Wheeler, Genbank, *Nucleic Acids Res.* **27** (1999), 12.

[7] J.A. Brzozowski, Derivatives of regular expressions, *Journal of the ACM* **11** (1964), 481.

[8] S.V. Buldyrev, A.L. Goldberger, C.K. Peng, M. Simons and H.E. Stanley, Generalized levy-walk model for DNA nucleotide sequences, *Phys. Rev.* **E47** (1993), 4514.

[9] A. Czirok, R.N. Mantegna, S. Havlin and H.E. Stanley, Correlations in binary sequences and a generalized Zipf analysis, *Physical Review* **E52** (1995), 446.

[10] W. Ebeling and G. Nicolis, Word frequency and entropy of symbolic sequences: A dynamical perspective, *Chaos, Solitons and Fractals* **2** (1992), 635.

[11] I. Foster, C. Kesselman and S. Tuecke, The anatomy of the grid: Enabling scalable virtual organizations, *Intl J. Supercomputer Applications* **15** (2001), 200.

[12] D.G. Higgins, A.J. Bleasby and R. Fuchs, Clustal V: Improved software for multiple sequence alignment, *CABIOS* **8** (1992), 189.

[13] S. Karlin, B.E. Blaisdell, R.J. Sapolsky, L. Cardon and C. Burge, Assessments of DNA inhomogeneities in yeast chromosome III, *Nucleic Acids Res.* **21** (1993), 703.

[14] S. Karlin and V. Brendel, Chance and statistical significance in protein and DNA sequence analysis, *Science* **257** (1992), 39.

[15] S. Karlin and V. Brendel, Patchiness and correlations in DNA sequence, *Science* **259** (1993), 677.

[16] P. Katsaloulis, T. Theoharis and A. Provata, Statistical distributions of oligonucleotide combinations: Applications in human chromosomes 21 and 22, *Physica* **A316** (2002), 380.

[17] K. Levenberg, A method for the solution of certain problems in least squares, *Quart. Appl. Math.* **2** (1944), 164.

[18] R.N. Mantegna, S.V. Buldyrev, A.L. Goldberger, S. Havlin, C.K. Peng, M. Simons and H.E. Stanley, Linguistic features of noncoding DNA sequences, *Phys. Rev. Letts.* **73** (1994), 3169.

[19] D. Marquardt, An algorithm for least-squares estimation of nonlinear parameters, *SIAM J. Appl. Math.* **11** (1963), 431.

[20] Nomenclature Committee of the International Union of Biochemistry (NCIUB), Nomenclature for incompletely specified bases in nucleic acid sequences. recommendations 1984, *Biochem. J.* **229** (1985), 281.

[21] Nomenclature Committee of the International Union of Biochemistry (NCIUB), Nomenclature for incompletely specified bases in nucleic acid sequences. recommendations 1984, *Eur. J. Biochem.* **150** (1985), 1.

[22] Nomenclature Committee of the International Union of Biochemistry (NCIUB), Nomenclature for incompletely specified bases in nucleic acid sequences. recommendations 1984, *J. Biol. Chem.* **261** (1986), 13.

[23] W.R. Pearson and D.J. Lipman, Improved tools for biological sequence comparison, *Proc Natl Acad Sci USA* **85** (1988), 2444.

[24] C.K. Peng, S.V. Buldyrev, A.L. Goldberger, S. Havlin, F. Sciortino, M. Simons and H.E. Stanley, Long-range correlations in nucleotide sequences, *Nature* **356** (1992), 168.

[25] W.H. Press, S.A. Teukolsky, W.T. Vetterling and B.P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge Univ. Pr., 1993.

[26] A. Provata and Y. Almirantis, Scaling properties of coding and non-coding DNA sequences, *Physica* **A247** (1997), 482.

[27] P.R. Tome, P.J. Stoehr, G.N. Cameron and T.P. Flores, The european bioinformatics institute (EBI) databases, *Nucleic Acids Res.* **24** (1996), 6.

[28] A.A. Tsonis, J.B. Elsner and P.A. Tsonis, Periodicity of DNA coding sequences: Implications in gene evolution, *J. Theor. Biol.* **151** (1991), 323.

[29] http://gcc.gnu.org/.

[30] http://http.ncbi.nlm.nih.gov/GenBank.

[31] http://plasma-gate.weizmann.ac.il/Grace/.

[32] http://www.bioinfo.gr.

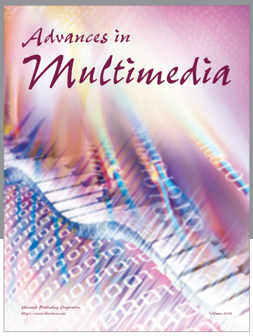
[33] http://www.ebi.ac.uk/fasta33.

[34] http://www.edi.ac.uk/clustalw.

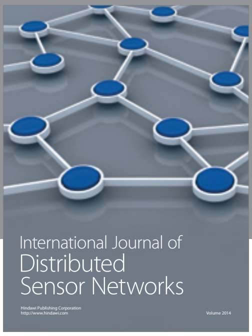
[35] http://www.gnu.org/software/bash/bash.html.

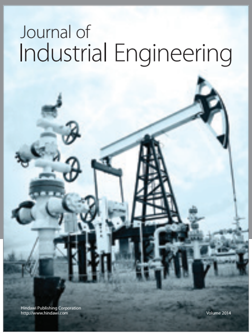
[36] http://www.gnuplot.info/docs/gnuplot.html.

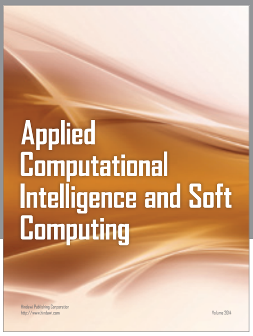
[37] http://www.ncbi.nlm.nih.gov/BLAST.

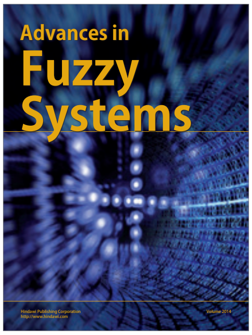

Advances in
Multimedia

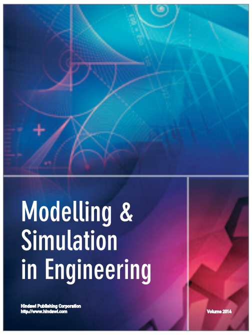
The Scientific
World Journal

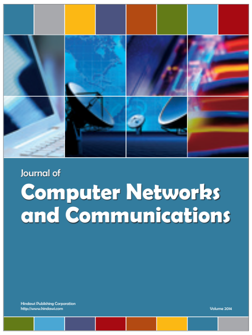

International Journal of
Distributed
Sensor Networks

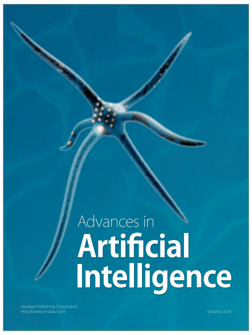

Journal of
Industrial Engineering

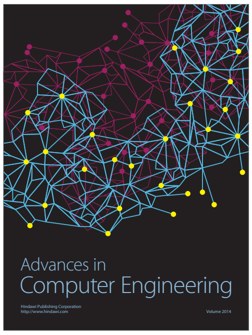

Applied
Computational
Intelligence and Soft
Computing


Advances in
Fuzzy
Systems

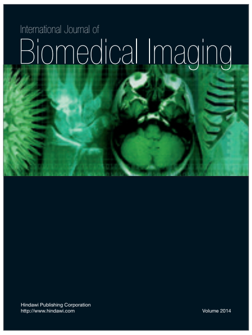

Modelling &
Simulation
in Engineering

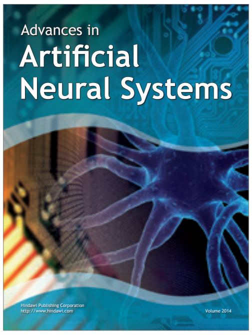

Journal of
Computer Networks
and Communications

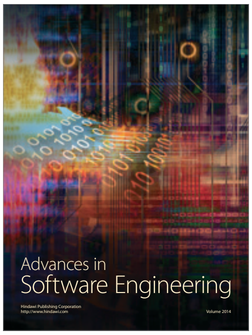

Advances in
Artificial
Intelligence


Advances in
Computer Engineering

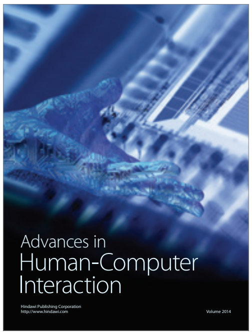
International Journal of
Computer Games
Technology

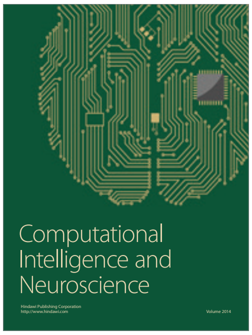

International Journal of
Biomedical Imaging


Advances in
Artificial
Neural Systems


Advances in
Software Engineering

Journal of
Robotics


Advances in
Human-Computer
Interaction


Computational
Intelligence and
Neuroscience

International Journal of
Reconfigurable
Computing

Journal of
Electrical and Computer
Engineering