BIOLETS: STATISTICAL APPROACH TO BIOLOGICAL RANDOM SEQUENCE GENERATION

Vinod Chandra S. S[1] , Gopakumar G[2] and Achuthsankar S. Nair[2]
[1]Department of Computer Science and Engineering, College of Engineering, Thiruvananthapuram-695016, Kerala, India, vinodchandrass@gmail.com
[2]Centre for Bioinformatics, University of Kerala, Thiruvananthapuram-695581, Kerala, India, gopakumar.cbi@gmail.com, sankar.achuth@gmail.com

ABSTRACT

Simulations of originally existent biological sequences are helpful for computational analysis and manipulations, rather than wet lab experiments, considerably reducing the cost and time associated with traditional wet lab experiments. The basic idea is to compare the results of a run on real data to many runs on random data. This paper discusses the need for random biological sequence analysis and an alternative for generating biological random sequences. A novel random generation algorithm which uses the statistical distributions to generate numbers and to map these numbers into biological random sequences is conceived. Algorithm for generating random sequences with normal and binomial distributions is implemented; random sequences with tandem repeats, GC base control and nucleotide position control are also incorporated by modifying this algorithm. The newly proposed algorithm and all the mentioned features are also implemented as a platform independent tool named Biolets which can be freely downloaded from the web site http://sooryakiran.com/products_biolets.html.

Keywords: Biological Sequence Generation, Random DNA Sequence, Statistical Distributions, Sequence Analysis, Tandem Repeats.

1.0 INTRODUCTION

Bioinformatics is generally defined as the analysis, prediction, and modeling of biological data with the help of computers. It is the application of computer science and allied technologies to answer the questions about the mysteries of life and is mainly concerned with problems involving data emerging from within cells of living beings [1]. The most important data in the bioinformatics domain are the sequences of DNA, RNA and Proteins. DNA sequencing technologies have created massive amounts of information that can only be efficiently analyzed with computers [2, 3, 4]. As the information becomes ever so larger and more complex, more computational tools are needed to sort through the data.

Computational analysis of biological sequences has become an invaluable tool in modern molecular biology. Some of the applications are aligning sequences or sets of sequences, recognizing coding regions of DNA, prediction of protein secondary structure, detecting relationships among genes, proteins or species and constructing evolutionary trees [5, 6]. In all the above applications the question of the statistical significance of the results is one of the most important and difficult to address. For example if two DNA sequences are found to share a common subsequence of a certain length, does it imply that the two sequences are functionally or evolutionarily related? The answer to this is very much depending on the finding obtained. Since the alphabet of proteins consists of amino acids and the alphabet of DNA is the nucleotides A, G, C and T, certain repeats can or must occur by chance. The significance of any finding must therefore be judged relative to a background level expected by chance alone.

It is very difficult to obtain the mathematical results concerning significance level for sequence analysis findings and is known only in some special situations. Therefore simulations are often used to provide a statistical background. The basic idea is to compare the results of a run on real data to many runs on random data. The difficulty to be addressed here is how to construct an appropriate random data. For example, note that two DNA sequences that are rich in G and C nucleotides are more likely to have a common subsequence of a given length than two sequences in which the nucleotides are equally frequent. However, the mere fact that the two sequences are G, C rich might be of some interest which leads us to ask the next question. Given that the two sequences are G, C rich, what is the significance of finding a certain common subsequence? To answer this question it is natural to

create random sequences that have the same nucleotide composition as the original sequences. If the objective is merely to produce a uniformly random sequence with the same number of As, Gs, Cs and Ts as a given sequence, there are two simple efficient procedures available. One can either tally the frequencies in the given sequence to generate a uniform permutation of the nucleotide multi-set or one can shuffle the given sequence until it is adequately mixed.

Random sequences can be used to extract relevant information from biological sequences [7]. The random sequences represent the 'background noise' from which it is possible to differentiate the real biological information. Random sequences are widely used to detect over-represented and under-represented motifs [8, 9], or to determine whether the scores of pair wise alignments are relevant. Some programs, currently available for generating random sequences, use techniques like Markov chains, Hidden Markov Models [10], weighted context-free grammars, regular expressions etc [7]. In all of the above types of random sequence generation, homogenous model of the probability of a letter depends on the previous letters in the sequence.

There exist some systems available for random sequence generation and are standalone or online tools with focus on either only one type of generation or one type of sequence. GenRGenS is a software tool dedicated to randomly generating genomic sequences [7]. This tool handles several classes of model useful for sequence analysis such as hidden Markov models, weighted context-free grammars and regular expressions. *RSA* (Random Sequence Generator) tool [11] generates random DNA sequences by a probabilistic model. *RandSeq* is a tool used for random protein sequence generation. *RANDNA* [12, 13] is a random DNA sequence generator.

All these are standalone or online tools that support only one kind of biological random sequence generation. It would be more useful if there is a tool that can generate all types (DNA, RNA & Protein) of random biological sequences. Moreover, there is no such a tool that allows user to control the percentage of occurrence of nucleotides or GC base pair positions or tandem repeating units in the sequence to be generated. The application program developed as a prototype of this paper can handle the generation of all the three kinds of sequences viz. DNA, RNA and Proteins, using statistical distributions. This tool also incorporates the mentioned unique features giving user some control over the random sequence to be generated as per his/her wish.

2.0 MATERIALS AND METHODS

In the present work, random numbers sequences are generated using statistical distributions followed by testing these numbers' probability of falling within a normal distribution curve. Each such valid random number generated is then mapped into corresponding, previously set, nucleotides in the case of DNA/RNA sequences or amino acids in the case of Protein sequences. The choice of type of the sequence, length of the sequence, distribution of nucleotides (in most cases with exception to binomial distribution) and generation constraints need only be specified as prerequisites. The methodology used in this work for random sequence generation can be summarized as**:**

- Identify the type of sequence and other features set by the user.
- For normal distribution and uniform distribution of probabilities of bases/amino acids, deploy the uniform distribution curve method to generate random numbers and map these numbers into nucleotides.
- For probabilities where base pairs are to be generated, deploy the binomial distribution to generate random numbers, followed by mapping into nucleotides/amino acids.

For discrete random variables (i.e. if the random variables assume non-zero values), we have the probability distribution function as f(x) where x is the random variable. Here x takes a value between 0 and 1. The uniform distribution function takes the value of ¼, giving equal probabilities to all the bases. This means that the probability of occurrence of any one base at a particular position inside the sequence length is ¼. This is decided by the random variable generated each time by the system. For the generation of random variable we use random system calls and use uniform distribution to deploy the bases in the sequence. The algorithm takes a slight modification where pair wise permutation is needed to be considered for the sequence generation. For example, in the case of purines, combined percentage of nucleotides A and G will be input and a binomial distribution will be used to distribute them in the given sequence, the same procedure is applied for pyrimidines also.

Binomial distributions are used in cases where we want to generate base pairs as in the case of generating sequence by specifying the purines – pyrimidines percentage. Here each pair of purines is treated as a single entity and it assigned the value of either 1 or 0. Once random value of purines is fixed, pyrimidines take the compliment value of

0 or 1. The same method of generating and distributing random numbers as in the case of uniform distribution can be employed here also with a modification that the probability function gives the value of ½. The same method is deployed in distributing the bases within the purines – pyrimidines region. The algorithm used for generation is:

1. Make the choice of sequence and enter the length of sequence.
2. Specify the frequencies of nucleotides.
3. Generate the binomial distribution curve with length constraints.
4. Map the curve to permutation of nucleotide multi-set.

After generating the random sequence of interest, the energy value associated with each of the sequence generated can be calculated. In Thermodynamics, the Gibbs free energy G describes the energy of molecules in aqueous solution. The change ΔG of the free energy in a chemical process, such as nucleic acid folding, determines the direction of the process. The free energy of the secondary structure is calculated using the Nearest Neighborhood Rule [14]. The Nearest Neighbor hood rule states that the energy of the structure is the total energy of the Watson-Crick base pairs in the structure [15]. Each base pair has constant free energy as given below in Table 1.

Table 1: Watson- Crick base pair energy

INTERACTION	**ΔG VALUE**
AA / TT	1.9
AT/TA	1.5
TA/AT	0.9
CA/GT	1.9
GT/CA	1.3
CT/GA	1.6
GA/CT	1.6
CG/GC	3.6
GC/CG	3.1
GG/CC	3.1

To calculate the free energy associated with the DNA sequence, we sum up the predicted free-energy values (kcal/mol at 25° C) for base pair stacking. In this tool, values from the given table are taken and the free energy value associated with each random sequence of DNA is calculated and reported with the sequence. Other than simply generating the random sequence of interest with free energy value, biological random sequences with advanced features like position control, GC base pair control, tandem repeats and supersede control are also generated in this tool as given in the following sub sections.

2. 1 Third Position Control

The genetic code is the set of rules by which information encoded in genetic material (DNA or RNA sequences) is translated into proteins (amino acid sequences) in living cells. Specifically, the code defines a mapping between tri-nucleotide sequences called codons [16] and amino acids; every triplet of nucleotides in a nucleic acid sequence specifies a single amino acid. Because the vast majority of genes are encoded with exactly the same code, this particular code is often referred to as the canonical or standard genetic code. There are $4^3 = 64$ different codon combinations possible with a triplet codon of three nucleotides. In reality, all 64 codons of the standard genetic code are assigned for either amino acids or stop signals during translation. Since the same four bases code for the proteins, the third position base in each triplet is of much importance. It is this third position base that determines which amino acid to be coded. If, for example, a DNA sequence, GTATCGTATCCGTAC, is considered and the reading-frame starts with the first G. The third codon TAT and that last codon TAC will code for the same amino acid Threonine (T). It is to be noted in this context that the other two codons (TAG and TAA) whose third position only varies compared to TAT and TAC are stop codons. Stop codons are also called termination codons and they

signal release of the nascent polypeptide from the ribosome. From the example the significance of the third position base in each codon could be easily understood. The proposed system is capable of controlling the third position base in each of the sequence generated by declaring each third position a reserved area for the nucleotide. This is completely dependent on the type of the sequence and the user's choice of the third position base.

2. 2 GC Base Pair Control

This may refer to a specific fragment of DNA or RNA, or that of the whole genome. When it refers to a fragment of the genetic material, it may denote the GC-content of part of a gene (domain), single gene, group of genes (or gene clusters) or even a non-coding region. In the proposed system, a DNA/RNA sequence with a specific fragment of the code with maximum probability to GC base pairs is generated. This is accomplished by setting a GC window and there by generating the fragment within the GC window with maximum probability to the GC bases. Again the filling up of the GC window takes place in a random manner. This feature is applicable only for the DNA / RNA sequence.

2. 3 Tandem Repeats

Tandem repeats and variable number tandem repeats in DNA occur when a pattern of two or more nucleotides is repeated and the repetitions are directly adjacent to each other. Tandem repeats can be very useful in determining parentage. Short tandem repeats are used for certain genealogical DNA tests. It is a class of polymorphisms that occurs when a pattern of two or more nucleotides are repeated and the repeated sequences are directly adjacent to each other. As mentioned, in the DNA sequence GTATCGTATCCGTAC, the tandem repeat sequence GTATC is repeated once. User should specify the tandem sequence pertaining to some preconditions and on positive test of the conditions; the tandem sequence will be repeated in the random sequence by number of times specified by the user. The deployment of the tandem sequence is also done in the random fashion. That is, the position of the tandem window changes, randomly, for each generation. Tandem sequence generation is another feature which is available only on user choice of sequence types DNA / RNA.

2. 4 Supersede Control

In most of the analysis and manipulation purposes of the biological sequences, we would want random sequences where, the occurrence and positions of the nucleotides in the sequences are relevant. One such typical case is the third position control mentioned above. Supersede control is another mechanism where by random biological sequences are generated by controlling the preceding and superseding bases. The method of implementation is similar to the one deployed in third position control, but with the exception that we are interested in controlling every alternating bases in the sequence.

3. 0 RESULTS AND DISCUSSIONS

This paper analyses statistical approach of biological random sequence generation. As a part of current work, a tool named *Biolets* has been developed based on the statistical method explained. The tool can be freely downloaded from the web site http://sooryakiran.com/products_biolets.html. The tool is capable of generating biological random sequences of DNA, RNA and Protein. This feature is unique to *Biolets* compared to other tools which were able to generate only one type of sequence. The tool also comes with four other unique features mentioned above viz third position control, GC base pair control, tandem repeats and supersedes control. User can generate random DNA/RNA sequences with these features incorporated. There is also option in *Biolets* to generate genomic sequences with user defined probability of nucleotides (e. g, A=0.35, C=0.25, G=0.3 & T=0.1). Biolets also allows user to specify only purine percentage and generates the sequence by finding pyrimidines percentage, using a binomial distribution.

The feature third position control of codons helps user to manually set a particular amino acid in the sequence to be generated. In the case of GC base pair control feature, there is option in *Biolets* to specify the starting and ending positions in the random sequence to be generated. This gives more control for user in the sequence generation. The tandem repeat feature of *Biolets* allows user to specify the block of nucleotides to be repeated and the number of times that particular sun sequence to be repeated in the sequence to be generated. In the supersede control feature, Biolets allows user to specify the nucleotide which should come after the every occurrence of a particular nucleotide

which can also be set by the user. This feature also gives the user more control over the sequence to be generated. All these features of *Biolets* make it a unique tool in the area of biological random sequence generation than the existing ones. These novel features of *Biolets* will surely boost research associated with random genomic and proteomic sequences.

The study conducted so far focuses on trivial sequence features explained above and surely there is room for further expansion. Incorporation of features like:

- Incorporation of Inverted Repeats, SINE and LINE,
- Secondary structure prediction of RNA and calculation of fold energies associated with protein sequences could also be implemented, and
- Energy calculations at various physical conditions could be incorporated

will definitely make *Biolets* a very useful tool for biological sequence analysis and related studies like synthetic biology.

REFERENCES

[1] N. M. Luscombe, D. Greenbaum and M. Gerstein, "Bioinformatics: A Proposed Definition and Overview of the Field", *Method of Information in Medicine*, Vol. 4, 2001, pp. 346 – 358.

[2] J. Skolnick and J. S. Fetrow, "From Genes to Protein Structure and Function: Novel Applications of Computational Approaches in the Genomic era", *Trends Biotechnology*, Vol. 18, 2000, pp. 34 - 39.

[3] A. A. Mironov, E. V. Koonin, M. A. Roytberg and M. S. Gelfand, "Computer Analysis of Transcription Regulatory Patterns in Completely Sequenced Bacterial Genomes", *Nucleic Acids Research,* Vol. 27, No. 14, 1999, pp. 2981-2989.

[4] M. S. Gelfand, "Prediction of function in DNA sequence analysis", *Journal of Computational Biology,* Vol. 1, 1997, pp. 87-115.

[5] S. Karlin and V. Brendel, "Chance and Statistical Significance in Protein and DNA Sequence Analysis", *Science*, Vol. 257, 1992, pp. 39 -49.

[6] E. M. Marcotte, M. Pellegrini, H. L. Ng, D. W. Rice, T. O. Yeates and D. Eisenberg, "Detecting protein function and Protein-Protein Interactions from Genome Sequences", *Science*, Vol. 285, 1999, pp. 751-3.

[7] Yann Ponty, Michel Termier and Alain Denise, "GenRGenS: Software for Generating Random Genomic Sequences and Structures", *Bioinformatics*, Vol. 22, No. 12, 2006, pp. 1534-1535.

[8] A. M. McGuire, J. D. Hughes and G. M. Church, "Conservation of DNA Regulatory motifs and Discovery of new Motifs in Microbial Genomes", *Genome Research*, Vol. 10, No. 6, 2000, pp. 744-757.

[9] John N Towse and Andrea Cheshire, "Random Number Generation and Working Memory", *European Journal of Cognitive Psychology*, Vol. 19, No. 3, 2007, pp. 374 – 394.

[10] E. Birney, "Hidden Markov models in Biological Sequence Analysis", *IBM Journal of Research and Development*, Vol. 45, 2001, pp. 449-455.

[11] P. Flajolet, Paul Zimmerman and B. V. Cutsen, "A Calculus for the Random Generation of Labeled Combinatorial Structures". *Theoretical Computer Science*, *Vol.* 132, No. 1-2, 1994, pp. 1–35.

[12] A. Rambaut and N. C. Grassly, "Seq-Gen: an Application for the Monte Carlo Simulation of DNA Sequence Evolution along Phylogenetic Trees", *Bioinformatics*, Vol. 13, No. 2, 2004, pp. 235–238.

[13] Francesco Piva and Giovanni Principato, "RANDNA: A Random DNA Sequence Generator", *InSilico Biology*, Vol. 6, 2006, pp. 253 – 258.

[14] N. M. Luscombe, S. E. Austin, H. M. Berman and J. M. Thornton, "An overview of the Structures of Protein-DNA Complexes", *Genome Biology*, Vol. 1, No. 1, 2000, pp. 1-37

[15] M. Sundaralingam and P. K. Ponnuswamy, "Stability of DNA Duplexes with Watson-Crick Base Pairs: A Predicted Model", *Biochemistry*, Vol. 43, No. 51, 2004, pp. 16467 -16476.

[16] S. F. Altschul and B. W. Erickson, "Significance of nucleotide Sequence Alignments: A Method for Random Sequence Permutation that Preserves Dinucleatide and Codon Usage", *Molecular Biology Evolution,* Vol. 2, 1985, pp. 526 – 538.

ACKNOWLEDGEMENTS

We thank Anand Sadasivan, Department of Computer Applications, College of Engineering Trivandrum for his help and support. We also thank SooryaKiran Bioinformatics (P) Ltd (www.sooryakiran.com), Nila, Technopark, Thiruvananthapuram – 695 581 for the support provided.

BIOGRAPHY

Vinod Chandra S. S. is presently working as a faculty member in Department of Computer Science & Engineering, College of Engineering Thiruvananthapuram, Kerala. Presently he is doing Ph. D. in Computational Biology in University of Kerala. He has a modest number of research publications in National and International levels.

Gopakumar G. is a Doctoral student at Centre for Bioinformatics, University of Kerala, India. His areas of interest include fractal analysis of biological sequences, biological sequence analysis and protein-protein interactions.

Achuthsankar S. Nair heads the Centre for Bioinformatics, University of Kerala, India. He holds a B.Tech and M.Tech from the College of Engineering, Trivandrum and IIT Bombay, respectively, both in electrical engineering. He also holds an M.Phil. in computer speech and language processing from the University of Cambridge, UK, and a Ph.D. from the University of Kerala.