7

Bioinformatics: strategies, trends, and perspectives

Carlos Norberto Fischer and Adriane Beatriz de Souza Serapião
São Paulo State University – UNESP – Rio Claro, S.P.
Brazil

1. Introduction

With the advances in the genome area, new techniques and automation processes for DNA sequencing, the amount of data produced has increased exponentially. Analyzing this data, in order to identify interesting biological features, is an enormous challenge, especially if it would be done manually. Think about trying to find a specific word in a book, say Don Quixote, and we have to search word by word. How long it would take?

Bioinformatics has played an important role trying to help specialists to analyze data of a specific genome. The application of information technology, associated with techniques from applied mathematics, informatics, statistics, and computer science, has allowed the discovering of interesting and important characteristics in genomes, allowing to understand and solve several biological problems, or even to generate more knowledge or insight about the problem and its involved biological processes, what can bring advances in the used techniques.

In Computing area, for example, an ordinary type of task is to process texts. There are several problems involving strings, like trying to find a specific word (we could say "to align words") or a similar one (considering a particular pattern of characters) in a text. When processing genomic data, if it is desired to search for a specific pattern (and its approximations) in DNA sequences, the natural way is to use solutions already implemented. Thus, for pattern (exact or not) search and similar problems, bioinformaticians have developed computational tools that apply techniques and algorithms well-known in Computing area in order to solve these important genomic problems. Sometimes, they need to adapt algorithms for considering specific features of the biological problem. Two good examples of this case are Sequence Aligning and Sequence Assembly, processes resulting of adaptations in algorithms in order to consider insertion, deletion, and substitution of nucleotides in DNA sequences.

Some statistical and computational techniques, such as Hidden Markov Models (HMMs), Stochastic Grammars, and Conditional Random Fields (CRFs) have been successfully applied for modeling, analysis, discovery, classification, and alignment of biological sequences (Yoon & Vaidyanathan, 2004, 2005). HMMs (Rabiner, 1989) and Stochastic Grammars (Sakakibara et al., 1994) are forms of generative models to label sequences, assigning a joint probability distribution of, for example, the gene hidden structure y and the

observed nucleotide sequence *x*. The parameters of these methods are trained to maximize the joint likelihood of training examples. CRFs (Lafferty et al., 2001) are probabilistic frameworks for labelling and segmenting sequential data. They consist of an undirected graphical model that defines a single distribution over label sequences given a particular observation sequence.

An example of a very interesting biological problem is the one related to Transposable Elements (TEs). Computational techniques have been developed for systematic genome annotation of TE families, such as homology-based, structure-based, *de novo*, and comparative genomic methods (Bergman & Quesneville, 2007). Homology-based methods are supported on detecting similarity to known TE protein-coding sequences using prior knowledge acquired based on previously identified TE sequences (Agarwal & States, 1994; Kurtz et al., 2000). Structure-based methods bear prior knowledge about the common structural features shared by different TEs that are required in their mobility processes (Andrieu et al., 2004; McCarthy & McDonald, 2003). These two methods can include HMM, stochastic grammars, and CRF approaches. *De novo* method essays to find TEs using repeat regions in genome without considering prior information about TE structure or similarity (Bao & Eddy, 2002; Campagna et al., 2005; Edgar & Myers, 2005; Pevzner et al., 2004). Comparative genomic methods (Caspi & Pachter, 2006) seek for insertion regions where multiple alignments of genome sequences are disrupted by a large insertion in one or more species. Applying filtering and concatenating, insertion regions are then locally aligned by similarity with all other insertion regions to identify repeat insertion regions. However, all of these methods have presented limited success. Some available computational resources for TE discovery and TE detection are summarized in (Bergman & Quesneville, 2007).

We intend to present in the following sections a few examples of biological problems and solutions that can be used to attack them, from simple strategies to alternative ones.

2. Bioinformatics: Simple Solutions and Challenges

Depending on the features of the biological problem being treated, relatively simple solutions from, for example, Computing area can be sufficient for helping to solve it. Thus, in some situations, it is interesting, more immediate, and simpler to think in strategies that can consider the use of existent bioinformatics tools created for other purposes and not to solve the specific problem under investigation, avoiding the development of a new tool with functionalities similar to others. Next, we present two examples of this situation.

2.1 An Immediate Approach for Polymorphism Identification

Polymorphisms are related to the insertion, deletion, and substitution of one or more nucleotides in a DNA strand (Bell, 2002; Bentley, 2000), events that can occur during the duplication process of these strands. Two important and common types of polymorphism are the so called SNP - Single Nucleotide Polymorphism, which is related to only one nucleotide, and the InDels, when there are insertions and deletions of more than one nucleotide.

SNPs constitute until 90% of all human genetic variations and occur, in average, each 1000 pairs of nucleotide bases in human genome (Bentley, 2000). SNPs are very important genomic elements and can be related to several genetic diseases. Polymorphisms, in general, have received much interest from the specialists in genetic enhancement researches aimed at

plants and animals. They can be used as molecular markers and for species evolution studies.

When verifying alterations in sequences from different genomes, the natural way to detect SNPs/InDels is to do this in two separate steps: (a) to align the DNA sequences of interest and (b) to verify possible differences between them. A new bioinformatics tool can be developed for executing both steps in order to make this detection process more accurate and, mainly, faster.

However, a problem associated with the development of a new algorithm and its related computational tool is that all the implementation process can take a long period of time and the created tool can produce initially suspicious or unexpected results, demanding new hours of test execution and coding maintenance before the tool can be considered useful and trustful.

Thus, instead of developing a new computational program for aligning sequences (the first step inside the process described above for SNP/InDel identification), an immediate solution is to use already implemented and well-established computational tools aimed at aligning sequences. There are several bioinformatics tools for aligning sequences that can be used in processes of SNP/InDel detection; two examples are CAP3 (Huang & Madan, 1999) and Phrap (Phrap, 2009) assembler programs, well-known and largely used in several genomic projects, that align sequences and join them in groups making easier and faster the analysis process.

It is important to say that similar tools aimed to analyze the alignments must be developed in both cases, whether either a new tool is implemented for doing the alignments or an already existent assembler program is used.

This strategy of using an existent assembler for doing sequence alignments has been used for several research groups and inside complete applications aimed at SNP identification (for example, Matukumalli et al., 2006; Savage et al., 2005; Tang et al., 2006).

The Group of Bioinformatics of Rio Claro (GBIRC) has developed a computational system aimed to help specialists involved in SNPs (and InDels) analysis processes. We also have used the approach described above. For this, the system uses CAP3 for sequence alignment and our Perl scripts parse the results verifying possible differences between sequences. These scripts also extract all the information of interest from the results and insert them in a relational database (the use of a relational database facilitates information recovery and analysis of the results). Whether the number of putative polymorphisms detected in the process is great, the system can provide ways to do a selection of part of them based on specific criteria (for example, type of polymorphism, size, and location in a sequence), reducing the number of SNPs candidates to be analyzed. All this process is executed in an automated way. The system was implemented for the web platform, allowing collaborative work between remote researchers. The system and its DEMO can be found at http://gbirc.rc.unesp.br/snpcomp/.

2.2 Transposable Elements: A Challenge in Genomics

Repeat elements are a significant part of almost any genome. Repetitive sequences can be primary divided into three classes: local repeats (tandem repeats and simple sequence repeats), families of dispersed repeats (mostly Transposable Elements (TEs) and retrotransposed cellular genes), and segmental duplications (duplicated genomic fragments).

TEs are a heterogeneous class of genetic elements that vary in structure, mechanism of transposition, and choice of target sites. They are known by a variety of names, including controlling elements, cassettes, jumping genes, roving genes, mobile genes, and mobile genetic elements.

In recent years, much evidence has been found that repetitive sequences play a crucial role in various biological processes of eukaryotic genomes (Craig et al., 2002). Thus, the computational identification of repeats is currently receiving much attention. However, because of the mobile character, the traditional repeater finders cannot give satisfactory results in TE analysis and it is necessary more complex and robust models for this purpose.

The great importance of TEs for molecular mechanisms and genome evolution has generated increasing interest in developing methods and tools for more efficient, sensitive, and accurate discovering, classifying, detecting, and visualizing workflows of TEs in gene sequences. The amount and variety of TEs that can be found in a genome boast fundamental and huge challenges to genome sequencing, assembly, alignment, and annotation processes. The problem of repeat type classification is not well-defined (Kapitonov & Jurka, 2008; Wicker et al., 2007) and the development of computational algorithms to deal with it is still a hard task, despite some heuristic approaches have been attempted with some results. The main challenges for these approaches are to distinguish TEs from all other repeat classes and to identify distinct TE families. By turn, TE discovery and detection are influenced by other types of genomic repeats (simple repeats and segmental duplications).

Despite the TE classification used in projects around the world, some classes of TEs have remarkable features in their structures. Two examples are LTR class (TEs that present a Long Terminal Repeats) and TIR class (TEs presenting Terminal Inverted Repeats). For these cases, we could say, it is not difficult to produce computational applications for detecting these repeats - all is need is to search for repetitions, inverted or not, of character strings. But, for several other TE classes there is little (or even no) evident characteristic in their composition that could be used properly to construct computational tools.

However, for all TE classes, a simple solution can be used in order to overcome the lack of more evident characteristics. In some public databases, such as the NCBI GenBank (GenBank, 2009), the TIGR (Ouyang & Buell, 2004), and Repbase (RepBase, 2009), we can find a lot of sequences that have been characterized as TEs. These sequences can be downloaded and separated into classes. Thus, a simple approach to identify TEs can be considered: to align new (target) sequences against the known TE sequences, searching for similarity between them and classifying the target sequences as possible members of one of these classes.

In a similar way to SNP identification, we can develop a new bioinformatics tool for doing sequence alignments and verification of similarities. But, as described before, we have to consider the problems associated with the development of a new algorithm and its computational tool. Again, an immediate solution is to use already existent and well-established computational applications for doing both tasks. For this, a very useful tool is BLAST (Altschul et al., 1990). BLAST (Basic Local Alignment Search Tool) is a set of programs, largely used in so many genomic projects, that aligns a query sequence against known sequences of an available genomic database and reports their similarities using two metrics, called e-value and score, showing the alignments. Thus, BLAST can also be used for the identification of TEs inside a new set of genomic sequences. Another very useful bioinformatics tool that can be used for this purpose is RepeatMasker (Smit et al., 2009), an

application that searches for repeats using its associate database of sequences, including of TEs.

Several projects that intent to identify TEs have used these applications (BLAST and RepeatMasker). It is the case of our group. GBIRC has implemented SATEComp, a computational system aimed at helping specialists to analyze TEs. Initially, SATEComp used only BLAST. Although BLAST and RepeatMasker use similarities for identification of TEs, due to some reasons (preference of use, for completing the results produced by BLAST, etc.) we have opted for including the execution of RepeatMasker in our environment as well. Perl scripts developed by our group parse the results of both applications, extract all the information of interest and insert them in a relational database. The results are presented to the user in a tabular format facilitating the visualization and comparison of them. All this process (alignments with BLAST and RepeatMasker, extraction of information from the resulting files and insertion in the database) is executed automatically by SATEComp. This automatic computational pipeline allows a faster and more reliable identification of TEs in several genomes. Thus, our system can be used as a tool for performing comparative studies considering different organisms, task that would be quite complex and laborious if it was made by manual annotation procedures. The system was primarily constructed for the web platform. Thus, researchers remotely located can work in a collaborative way inside a same project. More about SATEComp can be found in (Fischer et al., 2008) and on Internet at http://gbirc.rc.unesp.br/satecomp/, that includes a DEMO of the system.

Another possible way to detect TEs in genomic sequences is through the use of Hidden Markov Models (HMMs). An HMM (Sean, 1998; Krogh, 1998) is a finite set of states, each of which is associated with a (generally multidimensional) probability distribution. Transitions among the states are governed by a set of probabilities called Transition Probabilities. In a particular state, an outcome or observation can be generated, according to the associated probability distribution.

HMMs have helped to solve some problems in molecular biology, in special as probabilistic profiles of protein families. An HMM can be used in order to search possible members of a particular protein family inside a new genomic database. One form of constructing an HMM for this purpose is to align known sequences of the family and capture information about each type of amino acid considering each column of the alignment (in other words, counting the number for each amino acid present in each position of the alignment). Thus, the HMM generated could be compared with each target biosequence.

Since HMMs can be used for identification of members of protein families, it is natural to think in applying them for detection of members of TE classes. Several groups have considered applying HMMs for doing this work (for example, Andrieu et al., 2004; Juretic, 2004; Urgi, 2009), again, including our group.

HMMER (Hmmer, 2009) is a computational application that has been successfully used for constructing HMMs and searching for known proteins in a new database. All that is necessary is to provide HMMER with the sequence alignments. Our group has executed several tests using HMMER with representative sequences of TEs. We observed that in some cases the results produced by HMMER were complementary to the ones generated by SATEComp. Thus, we decided to incorporate the HMM approach in our system in order to have more precise and reliable results inside the same computational environment.

However, there is a problem with this approach where we need representative sequences: new TE sequences can be lost in the analysis process due to the lack or weakness of

similarity between the target sequences and the sequences included presently into a class. Moreover, discovering of a new class of TEs is not possible using this approach because we must have the representative sequences of TEs for the alignments. Thus, other solutions for the construction of auxiliary tools in analysis processes of TEs, and to help in several other types of biological problem, must be proposed in order to obtain results more complete and reliable.

3. Recent Development of Natural Computing Techniques in Bioinformatics

Despite the great contributions Genomics and Proteomics have received from using algorithms and techniques developed in other areas, for many biological problems, the consideration of simple techniques is not enough due to the involved characteristics and conditions or the absence of additional knowledge about the problems. For these cases, it is necessary to develop new approaches or consider the application of more sophisticate already existent ones such as some solutions from Computational Intelligence area. Also, even for problems that have been already treated in Bioinformatics, there is a lack of more efficient, sensitive, and accurate tools. So, it is necessary to propose alternatives to treat a problem in a more efficient manner, developing strategies to attack it properly.

In last years, Natural Computing methods have been applied to many fields of Bioinformatics (Fogel, 2008; Hassanien et al., 2008; Masulli & Mitra, 2009), such as protein structure prediction, protein folding simulation, microarray data analysis, and gene regulatory networks modeling. Natural Computing is a branch of the Computational Intelligence area that extracts ideas from nature to develop computational systems for problem solving which is related to optimization, data processing, and analysis techniques. Among the many approaches within computing inspired by natural and biological principles, the most well-known ones are Neural Networks (Haykin, 1999), Fuzzy Systems (Pedrycz & Gomide, 2007), Evolutionary Computing (Michalewicz, 1998), Swarm Intelligence (Bonabeau et al., 1999), Immunocomputing (de Castro & Timmis, 2002), and Simulated Annealing (Kirkpatrick, 1983).

Neural Networks are motivated by the highly interconnected neural structures in the brain and the nervous system. Fuzzy Systems are based on an extension of traditional logic in order to represent uncertainty and qualitative reasoning. Evolutionary Computing (i.e., genetic algorithms, genetic programming, and grammatical evolution) uses concepts of mutation, recombination and natural selection from biology. Swarm Intelligence (i.e., including particle swarm optimization (Eberhart et al., 2001), ant colony systems (Dorigo & Blum, 2005), and bee hive algorithms (Baykasoglu et al., 2007)) mimics the emergent collective behaviour of groups of simple agents (like social insects, birds and fishes), that result from the local interactions of the agents with each other and with their environment. Immunocomputing (or artificial immune systems) is inspired by the principles and processes of the vertebrate immune system. Simulated Annealing stems from an analogy between the way in which a metal cools and freezes into a minimum energy crystalline structure. The main applications of these methods refer to tasks as signal processing, classification, clustering, feature selection, optimization, data visualization, data mining, and information fusion.

Protein structure prediction (PSP) and protein folding (PF) problems are currently some of the most interesting problems in molecular biology. Intelligent techniques have been

applied to these problems, like as Neural Networks (Pollastri et al., 2002), Evolutionary Computing (Pedersen & Moult, 1997; Cutello et al., 2006), Fuzzy Systems (Blanco et al., 2002), Immune Systems (Nicosia, 2004), Swarm Intelligence (Bahamish et al., 2008), and Simulated Annealing (Simons et al., 1997).

DNA sequence alignment is an important operation in Genomics and Proteomics that consists of finding similarity between genome segments. It is very useful, for instance, to study functional, structural or evolutionary relationships between organisms, to investigate gene regulation, and to predict the function of novel genes within any species. This task has also received contribution of these innovative techniques as Ant Colony Systems (Chen et al., 2007; Zhao et al., 2008; Chen et al., 2009) and Genetic Algorithms (Nguyen et al., 2002; Shyu et al., 2004; Jangam & Chakraborti, 2007).

The automatic motif discovery problem is a multiple sequence local alignment problem that involves the search for approximate matches. Motifs are conservative sequence patterns among the regulatory regions of correlated genes. A number of natural computing approaches have been proposed to handle the problem, including Neural Networks (Mahony et al., 2006), Genetic Algorithms (Hemalatha & Vivekanandan, 2008; Kaya, 2009; Venugopal, 2009), and Swarm Intelligence (Lei & Ruan, 2008).

Pattern discovery, DNA mapping, gene identification, and sequence labeling form challenging computational problems in Bioinformatics. Such problems can be treated with methods for data classification, detection, segmentation, clustering, or prediction. Again, Natural Computing approaches demonstrate to be useful for such purposes. Different techniques were used to accomplish these tasks, such as Neural Networks (Li et al., 2003; Mateos et al., 2003; You & Liu, 2007), Genetic Algorithms (Quesneville & Anxolabéhère, 2001; Pereira et al., 2009; Jacob et al., 2009), Immunocomputing (Wang et al., 2008a-b), Swarm Intelligence (Zhang et al., 2007; Greene et al., 2008), and Simulated Annealing (Filippone et al, 2005, 2006).

Detecting repetition by sequence alignment methods is relatively easy. However, TE identification problem is notoriously difficult because the processes of TE evolution remain questionable. The moving ability, coding capacity, duplicating process, and mutation dynamics of TEs are not well understood and are still debated. The integration of diverse computational tools and techniques for a comprehensive analysis of TEs is still an open challenge in Bioinformatics.

The problem of automated TE detection and classification seems to be deranged and jumbled. A systematic algorithmic procedure using the traditional methods appears to be unable to perform the covering of the variety of TEs' sequences and families. Thus, more adaptive procedures, including intelligent and versatile strategies, can furnish further improvements in accuracy of computational tools because different repeat compositions may demand adjusting of parameters in genome analysis.

Natural Computing is a promising framework for TE prediction, providing ability to capture dependencies and to incorporate non-probabilistic evidences and heterogeneous data. Nevertheless, despite of the great number of intelligent methods and approaches being used in several problems of the Bioinformatics field, the application of natural computing in TEs is still only bashfully investigated (Quesneville & Anxolabéhère, 2001; Langdon & Banzhaf, 2008). Our explanation makes contribution to Bioinformatics by calling attention to this important issue. More systematic studies of promoting natural computing as a whole for handling genome repeats are desirable.

On the other hand, traditional HMM can also take benefit of intelligent computing techniques to increase the model capability and improve its performance in different genomic problems. Some hybrid models were already developed incorporating Fuzzy Logic (Collyda et al., 2006, 2007; Bidargaddi et al., 2008), particle Swarm Optimization (Rasmussen & Krink, 2003), and Genetic Algorithms (Won et al., 2004) approaches.
There are several general concepts underlying many approaches in natural computing, like parallelism, adaptation, distributivity, interactivity, emergence, self-organization, feedback, etc. Yet, the employment of natural computing in genome problems goes far beyond such features. Regardless of applying a specific nature inspired computing method or using hybrid strategies with traditional methods to solve biological problems, there are other aspects related to this approach that can contribute to obtain models to both represent biological knowledge and predict the characteristics of biological systems, enhancing the quality and accuracy of solutions. Other aspect is the ability to capture interactions among data and variables. For these reasons, we believe that natural computing seems to offer fruitful contributions for this field and even to be extended to address problems not considered here.

4. Innovative Hardware and Architecture Solutions for Bioinformatics

Several research groups have dedicated much effort in trying to develop bioinformatics tools that would be executed faster than similar existent ones. There is no doubt that it is very important to develop new "faster" software.
However, it is necessary to propose solutions in terms of hardware as well, be as a dedicate and complex computational architecture or as a relatively simple but specialized device, in order to treat specifically genomic problems intending to reduce the processing time – for example a hardware aimed at DNA sequence comparison.
In the same way we described before for software, we present below some hardware and architecture solutions that can be considered with the objective of reducing the processing time and, consequently, the response time for tasks related to genomic analysis.
Maybe the most natural way to aim that objective is to execute several processes simultaneously, a very common form of obtaining faster answers. One example of hardware solution for reducing the response time is the one used by the National Center for Biotechnology Information - NCBI, USA. NCBI provides a public service where users can request searches in its molecular sequence databases with the BLAST tool. Daily, thousands of requests must be served, requiring high availability and high performance from the NCBI computers. A "simple" solution used by that center was to distribute BLAST searches across multiple worker nodes (as they say): each node is responsible for searches in only part of a database, reducing significantly the response time for the requests (Blast, 2009). Several other groups have applied known computational approaches similar to that used by NCBI, in special, the ones involving computer clusters and parallel processing.
On the other hand, some non-traditional computing approaches have been presented by other groups. For example, consider the following. The information generated in sequencing processes has been stored in computer files as sequences of nucleotides. Each sequenced DNA nucleotide can be represented as a character (A, C, G, T) and inside a computer as one byte, a set of eight bits. Since every bit can assume one of two values (zero or one), using eight bits, we can obtain 256 different combinations, that is, 256 distinct characters.

However, for a strand of DNA, it is necessary only four combinations (four characters), what can be represented using only two bits. With this approach, it is possible to fit four characters in each byte, reducing the size of each sequence file to one fourth of the original size. This approach can be used as a simple form of reducing the size of DNA databases and the space necessary in storage devices. Also, for several types of biological problems, less space in computer main memory will be required for processing data in that format. Moreover, since all the data set is smaller, a greater number of data (nucleotides) can be transferred between storage devices and main memory in each access, reducing the number of accesses necessary to those devices (that are considered slow) and, consequently, the transfer time.

However, probably the main benefit of this approach is that it can help in reducing the processing time of biosequences. Certainly, the most common operation when treating DNA sequences is comparison of nucleotides, which is used in searches for patterns, in sequence assembly processes, when aligning sequences, and in several other activities related to genomic sequences. In the current model of comparing two sequences, one nucleotide of a sequence is compared to one nucleotide of the other sequence. It means that one byte with one nucleotide is considered at a time in each comparison. Using the approach described above, four nucleotides into each byte of each sequence will be considered in each comparison, allowing to compare a greater quantity of nucleotides in each operation, reducing the total processing time.

Some works have been presented considering approaches similar to the described one. For example, Krishnamurthy and colleagues in (Krishnamurthy et al., 2007) describe about the design of BLASTN (the version of BLAST for nucleotides) for their architecture Mercury.

However, some (more) complex architectures have been proposed for reducing the processing time. For example, Hasan and colleagues in (Hasan et al., 2007) present an overview of various global and local sequence alignment methods describing about some considered architectures.

5. Conclusion

In this chapter we intended to present indications of forms to solve, or at least to help to solve, some important problems in Genomics. Depending on the characteristics of the biological problem being treated, simple and traditional algorithms from, for example, Computing area can be very useful to solve it. However, in many other situations, more sophisticate techniques must be considered in order to help to understand more complex problems and then to try to propose solutions, both in terms of software and dedicate hardware.

6. References

Agarwal, P. & States, D.J. (1994). The Repeat Pattern Toolkit (RPT): analyzing the structure and evolution of the *C. elegans* genome, *Proc Int Conf Intell Syst Mol Biol*, 2:1–9.

Altschul, S.F.; Madden, T.L.; Schäffer, A.A.; Zhang, J.; Zhang, Z.; Miller, W. & Lipman, D.J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs, *Nucleic Acids Res.*, Vol. 25, pp.3389-3402.

Andrieu, O.; Fiston, A-S.; Anxolabéhère, D. & Quesneville, H. (2004). Detection of transposable elements by their compositional bias, *BMC Bioinformatics*, Vol. 5:94.

Bahamish, H.A.A.; Abdullah, R. & Salam, R.A. (2008). Protein Conformational Search Using Bees Algorithm, *Proceedings of the 2th Asia International Conference on Modelling & Simulation*, pp. 911–916.

Bell, J.I. (2002). Single nucleotide polymorphisms and disease gene mapping, *Arthritis Res.*, Vol. 4 (Suppl. 3), pp. S273-S278.

Blanco, A.; Pelta, D.A. & Verdegay, J.-L. (2002). Applying a fuzzy sets-based heuristic to the protein structure prediction problem, *International Journal of Intelligent Systems*, Vol. 17, pp. 629–643.

Bao, Z. & Eddy, S.R. (2002). Automated *de novo* identification of repeat sequence families in sequenced genomes, *Genome Res.*, Vol. 112, pp. 1269–1276.

Baykasoglu, A.; Özbakır, L. & Tapkan, P. (2007). Artificial Bee Colony Algorithm and Its Application to Generalized Assignment Problem. In: *Swarm Intelligence: Focus on Ant and Particle Swarm Optimization*, F.T.S. Chan and M.K. Tiwari (Eds.), Itech Education and Publishing, Vienna, Austria, pp. 113–144.

Bentley, D.R. (2000). The Human Genome Project – An Overview, *Medicinal Research Reviews*, Vol. 20, pp. 189-196.

Bergman, C.M. & Quesneville, H. (2007). Discovering and detecting transposable elements in genome sequences, *Briefings in Bioinformatics*, Vol. 8, No. 6, pp. 382-392.

Bidargaddi, N.P.; Chetty, M. & Kamruzzaman, J. (2008). Hidden Markov models incorporating fuzzy measures and integrals for protein sequence identification and alignment, *Genomics Proteomics Bioinformatics*, Vol. 6, No. 2, pp. 98-110.

BLAST Documentation (2009). Available at: ftp://ftp.ncbi.nlm.nih.gov/blast/documents/blast-sc2004.pdf . Last access: May 2009.

Bonabeau, E.; Dorigo, M. & Theraulaz, G. (1999). *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press.

Campagna, D.; Romualdi, C.; Vitulo, N. et al. (2005). RAP: a new computer program for *de novo* identification of repeated sequences in whole genomes, *Bioinformatics*, Vol. 21, pp. 582–588.

Caspi, A.; Pachter, L. (2006). Identification of transposable elements using multiple alignments of related genomes, *Genome Res.*, Vol. 16, pp. 260-270.

Chen, L.; Liu, W. & Chen, J. (2007). Ant Colony Optimization Method for Multiple Sequence Alignment, *Proceedings of the 2007 International Conference on Machine Learning and Cybernetics*, Vol. 2, pp. 914-919.

Chen, W.; Liao, B.; Zhu, W.; Liu, H. & Zeng, Q. (2009). An Ant Colony Pairwise Alignment Based on the Dot Plots, *Journal of Comput Chem*, Vol. 30, pp. 93–97.

Collyda, C.; Diplaris, S.; Mitkas, P.A.; Maglaveras, N. & Pappas, C. (2006). Fuzzy Hidden Markov Models: a new approach in multiple sequence alignment, *Studies in Health Technology and Informatics*, Vol. 129, pp. 1245-1249.

Collyda, C.; Diplaris, S.; Mitkas, P.A.; Maglaveras, N. & Pappas, C. (2007). Enhancing the quality of phylogenetic analysis using fuzzy hidden Markov model alignments, *Studies in Health Technology and Informatics*, Vol. 124, pp. 99-104.

Craig, N.L.; Craigie, R.; Gellert, M. & Lambowitz, A.M. (2002). *Mobile DNA II*. American Society for Microbiology Press, Washington, D.C.

Cutello, V.; Narzisi, G. & Nicosia, G. (2006). A multi-objective evolutionary approach to the protein structure prediction problem, *Journal of The Royal Society Interface*, Vol. 3, pp. 139–151.

de Castro, L. & Timmis, J. (2002). *Artificial immune systems: A new computational approach*. Springer-Verlag, London. UK.

Dorigo, M. & Blum, C. (2005). Ant colony optimization theory: A survey, *Theoretical Computer Science*, Vol. 344, No. 2–3, pp. 243–278.

Eberhart, R.C.; Kennedy, J. & Shi, Y. (2001*). Swarm Intelligence*, Elsevier Science.

Edgar, R.C. & Myers, E.W. (2005). PILER: Identification and classification of genomic repeats, *Bioinformatics*, Vol. 21, pp. i152–i158.

Filippone, M.; Masulli, F. & Rovetta, S. (2005). Unsupervised gene selection and clustering using simulated annealing. In: Isabelle Bloch, Alfredo Petrosino, and Andrea Tettamanzi (eds.), WILF*, Lecture Notes in Computer Science*, Vol. 3849, Springer, pp. 229-235.

Filippone, M.; Masulli, F. & Rovetta, S. (2006). Supervised classification and gene selection using simulated annealing, *Proceedings of the Int. Joint Conf. on Neural Networks*, IEEE, pp. 3566-3571.

Fischer, C.N.; Cerri, R.; Costa, E.P. & Bacci Jr., M. (2008). SATEComp: a Tool for *in Silico* Analysis of Transposable Elements, *Proceedings of World Academy of Science, Engineering and Technology*, Vol. 34, October, pp. 878-882.

Fogel, G.F. (2008). Computational intelligence approaches for pattern discovery in biological systems, *Briefings in Bioinformatics* , Vol. 9, pp. 307-316.

GenBank - NCBI (2009). Available at: http://www.ncbi.nlm.nih.gov/Genbank/ . Last access: May 2009.

Greene, C.S.; White, B.C. & Moore, J.H. (2008). Ant colony optimization for genome-wide genetic analysis, *Lecture Notes in Computer Science*, Vol. 5217, pp. 37-47.

HMMER Documentation (2009). Available at: http://hmmer.janelia.org/ . Last access: May 2009.

Hasan, L.; Al-Ars, Z. & Vassiliadis, S. (2007). Hardware Acceleration of Sequence Alignment Algorithms - An Overview, *IEEE Xplore*, Sept., pp.92-97.

Hassanien, A.-E.; Milanova, M.G.; Smolinski, T.G. & Abraham, A. (2008). Computational intelligence in solving bioinformatics problems: Reviews, perspectives, and challenges, *Studies in Computational Intelligence*, Vol. 151, pp. 3-47.

Haykin, S. (1999). *Neural Networks. A Comprehensive Foundation*, second ed., Prentice-Hall, Upper Saddle River, NJ.

Hemalatha, M. & Vivekanandan, K. (2008). Genetic algorithm based probabilistic motif discovery in unaligned biological sequences, *Journal of Computer Science*, Vol. 4, No. 8, pp. 625-630.

Huang, X. & Madan, A. (1999). CAP3: A DNA Sequence Assembly Program, *Genome Research*, Vol.9, n.9, pp. 868-877.

Jacob, E.; Nair, K.N.R. & Sasikumar, R. (2009). A fuzzy-driven genetic algorithm for sequence segmentation applied to genomic sequences, *Applied Soft Computing Journal*, Vol. 9, No. 2, pp. 488-496.

Jangam, S.R. & Chakraborti, N. (2007). A novel method for alignment of two nucleic acid sequences using ant colony optimization and genetic algorithms, *Applied Soft Computing*, Vol. 7, No. 3, pp. 1121-1130.

Juretic, N.; Bureau, T.E. & Bruskiewich, R.M. (2004). Transposable element annotation of the rice genome, *Bioinformatics*, Vol. 20, n.2, pp. 155-160.

Kapitonov, V.V. & Jurka, J. (2008). A universal classification of eukaryotic transposable elements implemented in RepBase, *Nature Reviews Genetics*, Vol. 9, pp. 411-412, May.

Kaya, M. (2009). MOGAMOD: Multi-objective genetic algorithm for motif discovery, *Expert Systems with Applications*, Vol. 36 (2 PART 1), pp. 1039-1047.

Kirkpatrick, S.; Gelatt, C.D. & Vecchi, M.P. (1983). Optimization by simulated annealing, *Science*, Vol. 220, pp. 661–680.

Krishnamurthy, P.; Buhler, J.; Chamberlain, R.; Franklin, M.; Gyang, K.; Jacob, J. & Lancaster, J. (2007). Biosequence Similarity Search on the Mercury System, Journal of VLSI Signal Processing Systems, Vol. 49, pp. 101-121.

Krogh, A. (1998). An Introduction to Hidden Markov Models for Biological Sequences, In: SALZBERG, Steven; SEARLS, David; KASIF, Simon (Comp.). *Computational Methods in Molecular Biology*. [s.i.]: Elsevier, Chap. 4, pp. 45-63.

Kurtz, S.; Ohlebusch, F.; Schleiermacher, C.; Stoye, J. & Giegerich, R. (2000). Computation and visualization of degenerate repeats in complete genomes, *Proc. Int. Conf. Intel. Syst. Mol. Biol*. 8: 228–238.

Lafferty, J.; McCallum, A. & Pereira, F. (2001). Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: *International Conference on Machine Learning*.

Langdon, W.B. & Banzhaf, W. (2008). Repeated patterns in genetic programming, *Natural Computing*, Vol. 7, pp. 589–613.

Lei, C. & Ruan, J. (2008). A particle swarm optimization algorithm for finding DNA sequence motifs, *Proceedings of the 2008 IEEE International Conference on Bioinformatics and Biomedicine Workshops* (BIBMW), art. no. 4686231, PA, USA, Nov., pp. 166-173.

Li, Y.; Yang, C. & Zhang, W. (2003). The Neural Network Method of DNA Sequence Classification, *Compute Simulation*, Vol. 20, No. 2, pp.65-68.

Mahony, S.; Benos, P.V.; Smith, T.J. & Golden, A. (2006). Self-organizing neural networks to support the discovery of DNA-binding motifs, *Neural Networks*, Vol. 19, pp. 950–962.

Masulli, F. & Mitra, S. (2009). Natural computing methods in bioinformatics: A survey, *Information Fusion*, Vol. 10, pp. 211–216.

Mateos, A.; Dopazo J.; Jansen, R.; Tu, Y.; Gerstein, M. & Stolovitzky, G. (2002). Systematic learning of gene functional classes from DNA array expression data by using multilayer perceptrons, *Genome Res*. Vol. 12, pp. 1703–1715.

Matukumalli, K.L; Grefenstette, J.J; , Hyten, D.L; Choi, I-Y; Cregan, P.B & Van Tassell, C.P. (2006). SNP-PHAGE - High throughput SNP discovery pipeline, *BMC Bioinformatics*, Vol. 7:468, Available from: http://www.biomedcentral.com/1471-2105/7/468.

McCarthy, E.M. & McDonald, J.F. (2003). LTR_STRUC: a novel search and identification program for LTR retrotransposons, *Bioinformatics*, Vol. 19, pp. 362–7.

Michalewicz, Z. (1998). *Genetic Algorithms + Data Structures = Evolution Programs*, Springer.

Nguyen, H.D.; Yoshihara, I.; Yamamori, K. & Yasunaga, M. (2002). Aligning multiple protein sequences by parallel hybrid genetic algorithm. *Genome Inform*., Vol. 13, pp. 123-132.

Nicosia, G. (2004). Immune *algorithms for optimization and protein structure prediction*, Ph.D. thesis, University of Catania, Italy.

Ouyang, S. & Buell, C.R. (2004). The TIGR Plant Repeat Databases: a collective resource for the identification of repetitive sequences in plants, *Nucleic Acids Research*, Vol. 32, Database issue, pp. 360-363.

Pedersen, J.T. & Moult, J. (1997). Protein Folding Simulations with Genetic Algorithms and a Detailed Molecular Description, *Journal of Molecular Biology*, Vol. 269, pp. 240-259.

Pedrycz, W. & Gomide, F. (2007). *Fuzzy Systems Engineering: Toward Human-Centric Computing*, IEEE/Wiley Interscience.

Pereira, P.; Silva, F. & Fonseca, N.A. (2009). BIORED - A genetic algorithm for pattern detection in biosequences, *Advances in Soft Computing*, Vol. 49, pp. 156-165.

Pevzner, P.A.; Tang, H. & Tesler, G. (2004). *De novo* repeat classification and fragment assembly, *Genome Res*., Vol. 14, pp. 1786–1796.

Phrap Documentation (2009). Available at: http://www.phrap.org/ . Last access: May 2009.

Pollastri, G.; Przybylski, D.; Rost, B. & Baldi, P. (2002). Improving the prediction of protein secondary structure in three and eight classes using recurrent neural networks and profiles, *Proteins*, Vol. 47, pp. 228–235.

Quesneville, H. & Anxolabéhère, D. (2001). Genetic algorithm-based model of evolutionary dynamics of class II transposable elements, *J. Theor. Biol*., Vol. 213, pp. 21-30.

Rabiner, L.R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition, *Proceedings of the IEEE*, Vol. 77, No. 2, pp. 257–285.

Rasmussen T.K. & Krink, T. (2003). Improved Hidden Markov Model training for multiple sequence alignment by a particle swarm optimization-evolutionary algorithm hybrid, *Bio Systems*, Vol. 72, No. 1-2, pp. 5-17.

RepBase (2009). Available at: http://www.girinst.org/server/RepBase/ . Last access: May 2009.

Sakakibara, Y.; Brown, M.; Hughey, R.; Mian, I.S.; Sjölander, K.; Underwood, R.C. & Haussler, D. (1994). Stochastic context-free grammars for tRNA modeling, *Nucleic Acids Res*., Vol. 22, pp. 5112-5120.

Savage, D.; Batley, J.; Erwin, T.; Logan, E.; Love, C.G; Lim, G.A.C.; Mongin, E.; Barker, G.; Spangenberg, G.C.; Edwards, D. (2005). SNPServer: a real-time SNP discovery tool, *Nucleic Acids Research*, Vol. 33, pp. 493-495.

Sean, R. E. (1998). Profile Hidden Markov Models. Bioinformatics Review, [s.i.], Vol. 14, n. 9, pp. 755-763.

Shyu, C.; Sheneman, L. & Foster, J.A. (2004). Multiple sequence alignment with evolutionary computation, *Genet. Prog. Evol. Mach*., Vol. 5, pp. 121-144.

Simons, K.T.; Kooperberg, C.; Huang, E. & Baker, D. (1997). Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions, *Journal of Molecular Biology*, Vol. 268, pp. 209–225.

Smit, A.F.A.; Hubley, R.; & Green, P. (2009) – RepeatMasker Documentation. Available at: http://www.repeatmasker.org . Last access: May 2009.

Szklarczyk, R. & Heringa, J. (2004). Tracking repeats using significance and transitivity, *Bioinformatics*, Vol. 20 (Suppl 1), pp. i311-i317.

Tang, J.; Vosman, B.; Voorrips, R.E.; van der Linden, C.G. & Leunissen, J.A.M. (2006). QualitySNP: a pipeline for detecting single nucleotide polymorphisms and insertions/deletions in EST data from diploid and polyploid species, *BMC Bioinformatics*, Vol. 7:438, Available from: http://www.biomedcentral.com/1471-2105/7/438.

Venugopal, K.R.; Srinivasa, K.G. & Patnaik, L.M. (2009). Merge based genetic algorithm for motif discovery, *Studies in Computational Intelligence*, Vol. 190, pp. 331-341.

Urgi (2009). Available at: http://urgi.versailles.inra.fr/research/index.php. Last access: May 2009.

Wang, L.; Zhang, J.; Huang, X.; Gong, G. (2008a). DNA sequences classification based on immune evolution network, *Proceedings of the 5th International Conference on Fuzzy Systems and Knowledge Discovery* (FSKD 2008), art. no. 4666157, pp. 448-452.

Wang, L.; Zhang, J.; Gong, G.; Peng, M. (2008b). Application of immune classifier based on increment of diversity in the model species genomes identification, *Proceedings of the International Conference on Intelligent Computation Technology and Automation* (ICICTA 2008), art. no. 4659437, pp. 30-35.

Wicker, T; Sabot, F.; Hua-Van, A.; Bennetzenm, J.L., Capy, P.; Chalhoub, B.; Flavell, A.; Leroy, P.; Morgante, M.; Panaud, O.; Paux, E.; SanMiguel, P. & Schulman, A.H. (2007). A unified classification system for eukaryotic transposable elements, *Nature Reviews Genetics*, Vol. 8, pp. 973-982, May.

Won, K.-J.; Prügel-Bennett, A. & Krogh, A. (2004). Training HMM structure with genetic algorithm for biological sequence analysis, *Bioinformatics*, Vol. 20, No. 18, pp. 3613–3619.

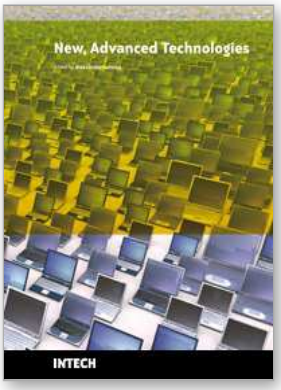
Yoon, B.-J. & Vaidyanathan, P.P. (2004). HMM with auxiliary memory: A new tool for modeling RNA secondary structures, *Proceedings of the 38th Asilomar Conference on Signals, Systems, and Computers*, Monterey, CA.

Yoon, B.-J. & Vaidyanathan, P.P. (2005). Optimal alignment algorithm for context-sensitive hidden Markov models, *Proceedings of the 30th International Conference on Acoustics, Speech, and Signal Processing*, Philadelphia.

You, W. & Liu, Y. (2007). Classifying DNA sequences by artificial neural network model, *Science and Technology Information*, Vol. 25, pp. 89-90.

Zhang, H.; Song, X. & Wang, H. (2007). Feature gene selection base on binary particle swarm optimization and support vector machine, *Computer and Applied Chemistry*, Vol. 24, No. 9, pp.1159-1162.

Zhao, Y.; Ma, P.; Lan, J.; Liang, C. & Ji, G. (2008). An improved ant colony algorithm for DNA sequence alignment, *Proceedings of the 2008 International Symposium on Information Science and Engineering* (ISISE 2008), Vol. 2, pp. 683-688.



New Advanced Technologies

Edited by Aleksandar Lazinica

ISBN 978-953-307-067-4
Hard cover, 350 pages
Publisher InTech
Published online 01, March, 2010
Published in print edition March, 2010

This book collects original and innovative research studies concerning advanced technologies in a very wide range of applications. The book is compiled of 22 chapters written by researchers from different areas and different parts of the world. The book will therefore have an international readership of a wide spectrum.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Carlos Norberto Fischer and Adriane Beatriz de Souza Serapião (2010). Bioinformatics: Strategies, Trends, and Perspectives, New Advanced Technologies, Aleksandar Lazinica (Ed.), ISBN: 978-953-307-067-4, InTech, Available from: http://www.intechopen.com/books/new-advanced-technologies/bioinformatics-strategies-trends-and-perspectives

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821