Chapter 7
Algorithmics for the Life Sciences

Raffaele Giancarlo

Abstract The life sciences, in particular molecular biology and medicine, have witnessed fundamental progress since the discovery of "the Double Helix". A relevant part of such an incredible advancement in knowledge has been possible thanks to synergies with the mathematical sciences, on the one hand, and computer science, on the other. Here we review some of the most relevant aspects of this cooperation, focusing on contributions given by the design, analysis and engineering of fast algorithms for the life sciences.

7.1 Introduction

In February 2001, the reference journals *Science* and *Nature* published special issues entirely dedicated to the sequencing of the human genome completed independently by *The Human Genome Consortium* and by *Celera Genomics*, with the use of two different sequencing approaches. Those results, of historic relevance, had already been widely anticipated and covered by the media since they are a fundamental landmark for the life sciences—biology and medicine, in particular. Indeed, the availability of the entire sequence of bases composing the human genome has allowed the comprehensive study of complex biological phenomena that would have been impossible before then. The abstraction process that allows a genome to be seen as a textual sequence is summarized in the box "Textual Representation of DNA". The race towards such a goal began in the early 1990s, when it became clear that the sequencing technologies available then, with the focused support of research in mathematics and computer science, could be extended to work on a genomic scale.

R. Giancarlo (✉)
Dipartimento di Matematica ed Informatica, Università di Palermo, Via Archirafi 34, 90123 Palermo, Italy
e-mail: raffaele@math.unipa.it

G. Ausiello and R. Petreschi (eds.), *The Power of Algorithms*,
DOI 10.1007/978-3-642-39652-6_7, © Springer-Verlag Berlin Heidelberg 2013

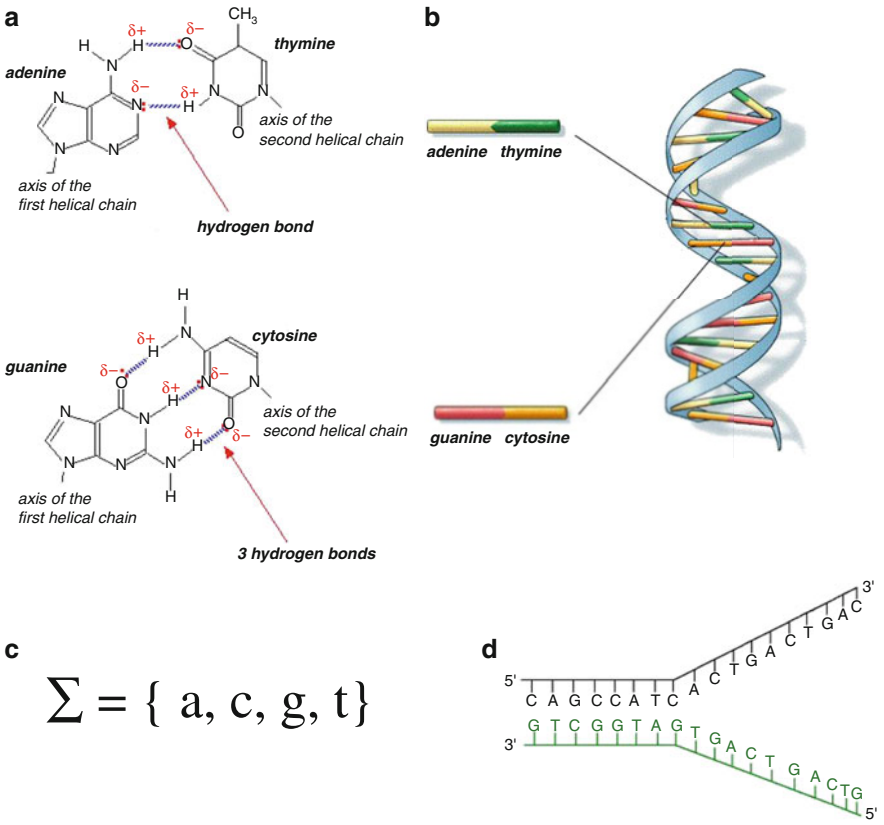

Fig. 7.1 A genome as a text sequence

Textual Representation of DNA

DNA can be represented in textual form, i.e., its biochemical structure can be described by a sequence of characters, as briefly outlined next. The four nucleic acids that compose DNA are Adenine, Cytosine, Guanine, and Thymine. They bind each other in a complementary way, as follows: Adenine and Thymine bind exclusively to each other and so do Cytosine and Guanine. The chemical structure of these bases and their links are illustrated in Fig. 7.1a. The "double helix" is shown in Fig. 7.1b. Its main features are: the skeleton of external support, made from sugars, represented on both sides as a ribbon, and the two filaments of bases linked in a complementary way that are represented as rods. The four nucleic acids can be simply "coded" with letters: A, C, G, T—with obvious association. The complementarity of

(continued)

(continued)

the bonds can be represented by pairs of letters (A, T) and (C, G). Once that is done, the biochemical structures of Fig. 7.1a can be represented by an alphabet of four symbols, shown in Fig. 7.1c. It is worth pointing out that a double-stranded DNA structure can be represented by choosing one of the two sequences corresponding to one of the two strands, since the other sequence can be uniquely determined from the chosen one (see Fig. 7.1d).

In the early 1990s, another technology with incredible potential gained the attention of biomedical research: that is, *microarrays*, which are chips that, intuitively, allow for the capture of information about genes that can be used to identify groups with common behavior in order to infer, for instance, the level of involvement of each group in the same pathologies. Almost simultaneously with the announcement of the completion of the human genome, have appeared in the same or in equally prestigious scientific journals, studies on the automatic classification of tumors, although there has not been much media coverage about them. The fundamentally new proposal in those studies is to produce accurate tumor taxonomies via gene expression experiments with the use of microarrays. Those taxonomies, in turn, are the initial point of research efforts that, in the future, would allow for the focusing of treatment of the specific pathology affecting a given patient. It is not a coincidence that microarrays are also a fundamental tool for drug design and discovery.

From the 1990s to today, thanks to the reduction in cost, both large-scale sequencing technologies and microarrays are part of the investigation tools of research institutions, even small ones. Such a widespread use of those so-called high-throughput technologies has resulted in data production in quantities such as to cause serious management problems both for data warehousing and analysis. Therefore, as a consequence, there has been an exponential growth both of specialized databases for biological research and of computer science tools essential for the analysis of those data.

Computer science, mathematics and statistics are therefore fundamental, certainly for the data warehousing aspects, but even more for the analysis of those data in order to reach conclusions of relevance for biological research. Algorithmics has already given contributions that are at the base of tools now recognized as essential for the life sciences, such as the program BLAST, which is of daily use for sequence database searches. Here an effort is made to give the reader an idea of those contributions, limiting the scope to some of the areas of the biomedical sciences where research is particularly intense and where the computational techniques have not yet reached their full potential. It has already been established, in the previous chapters, that algorithmics, tightly connected to combinatorics, proposes automatic procedures to determine solutions to many computational problems that are based on deep theories and that try to shed light on what is information and how it is best

represented and used. That will be exemplified in what follows by presenting some algorithms for bioinformatics that are recognized as reference points, either because they have been evaluated to be the best in a particular domain or because they have received particularly prestigious honors, such as the front page of outstanding scientific journals, e.g., *Nucleic Acids Research* and *BMC Bioinformatics*.

7.2 The Fundamental Machinery of Living Organisms

> A monkey is a machine that preserves genes up trees; a fish is a machine that preserves genes in water; there is even a small worm that preserves genes in German beer mats. DNA works in mysterious ways.[1]

DNA and proteins are polymers, composed of subunits known as nucleotides and amino acids, respectively. The genomic DNA of an organism, by means of the genes contained in it, is the information dictating the working of a complex biochemical machine whose aim is to produce proteins. Such DNA does not exist as a nude molecule, but rather as an extremely compact, three-dimensional, protein-DNA complex, known as chromatin. This latter is obtained via a process known as DNA supercoiling (Fig. 7.2), briefly described in box "DNA as Beads on a String". Intuitively, chromatin is the structure that is obtained once a long string has been wrapped around a series of beads in such a way as to take little space. The role of chromatin is not limited to such a compression process, as was initially thought, since it also has a deep influence on gene expression. Indeed, once packaged, only part of the genomic DNA is accessible and many messages are hidden. In order to transform those latter messages into proteins, they need to be made accessible. Such a goal is met via a dynamic behavior of the base components of chromatin. Here we limit ourselves to pointing out that research on such dynamic behavior is among the most important and fundamental in the life sciences because its understanding is seen as a substantial step forward for the cure of genetic diseases.

DNA as Beads on a String

Eukaryotic DNA can be seen as a very long thread. In order for this thread to be contained in the nucleus of a cell, it is necessary that it folds through a series of steps, at different levels of hierarchical organization, carried out through the use of particular proteins such as histones. The key steps of this process, known as supercoiling of DNA, are shown in Fig. 7.2. It is worth mentioning that only the first step of this process is known, while for the

(continued)

[1] Dawkins [25].

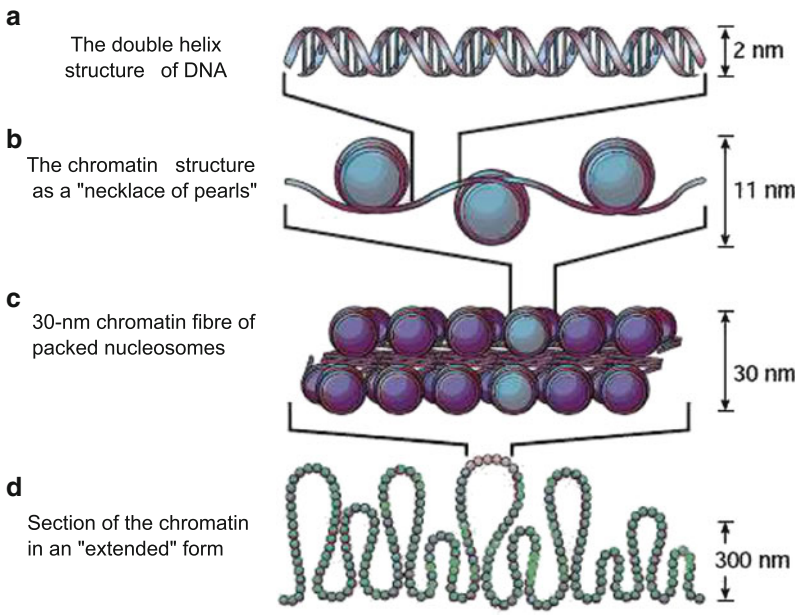

Fig. 7.2 The DNA supercoiling process

(continued)

others there are many working hypotheses about their structural conformation. The following explains in more detail the representations shown in the figure: (a) A DNA strand. The winding of DNA around spool-like structures called histones. Each spool with DNA wrapped around it is referred to as a nucleosome. (b) The resulting structure after this first step of packaging can be seen as a necklace of pearls, where the pearl is represented by the nucleosome. Note that the nucleosome is a fundamental unit in this process of compaction and bending, as DNA is wrapped at regular intervals, around histones to form nucleosomes. (c–d) The nucleosomes are then further packaged.

There is an estimate that each cell of a human being contains about 30,000 genes, that are present at birth and that remain stable throughout the entire life of an individual. Depending on various circumstances, including pathological ones or reactions to drugs, each gene is either activated or deactivated. In order for a gene to become active (technically, expressed), one needs to use "switchboards" referred to as promoters: DNA sequences that, on a genome, usually precede the DNA sequences corresponding to the genes. In each promoter, there are some "switches" that need to be "turned on" by some very special and important proteins,

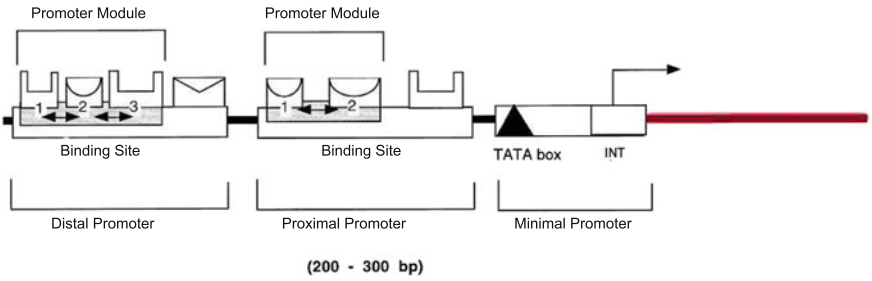

Fig. 7.3 The structure of a regulatory region

known as transcription factors. This entire process, known as transcription, is briefly illustrated next, starting with the description of the organization of a typical genomic region involved in the process, and schematized in Fig. 7.3. The region of interest can be divided into two smaller regions: "coding" (thick line) and "regulatory", where the first is to the right of the second. In the regulatory region, there are three promoters: the minimal promoter, very close to the start of the coding region, the proximal, which is further apart, and finally the distal, which can be even thousands of bases upstream of the beginning of transcription. That is not problematic since DNA is a dynamic three-dimensional structure and therefore sequences that are very far from each other may be close, or brought close to each other, in three-dimensional space. Some binding sites are indicated within each promoter.

When a binding site is "occupied" by a transcription factor, the effect is to recruit the RNA polymerase that begins the real transcription process. With reference to Fig. 7.4, the coding region is divided into introns and exons. The RNA polymerase transcribes the entire region forming the precursor RNA. Via a process known as splicing, some introns are removed to form the messenger RNA. This latter is then translated into an amino acid sequence corresponding to the desired protein.

The above transcription mechanism is common to all living species and it is therefore a fundamental one, whose malfunctioning may result in serious pathologies. Roger Kornberg, in 2006, received the Nobel Prize for Chemistry for his contributions to the understanding of the molecular basis of transcription. Those contributions could lead to the development of therapies, based on stem cells, for tumor and cardiovascular diseases.

The mechanisms and the processes that regulate gene expression and the quantity of proteins that result from that expression are extremely complex and much remains to be discovered and understood. One thing that is certainly clear is that malfunctioning of those expression mechanisms is at the origin of many pathologies. In order to shed light on the level of complexity of research in this area, we limit ourselves to mentioning that it has been discovered, only recently, that some small RNA sequences, known as microRNA, have a very important role in gene regulation, by inhibiting the production of some given proteins, de facto "silencing" or lessening the expression level of the corresponding genes. Recent studies, which

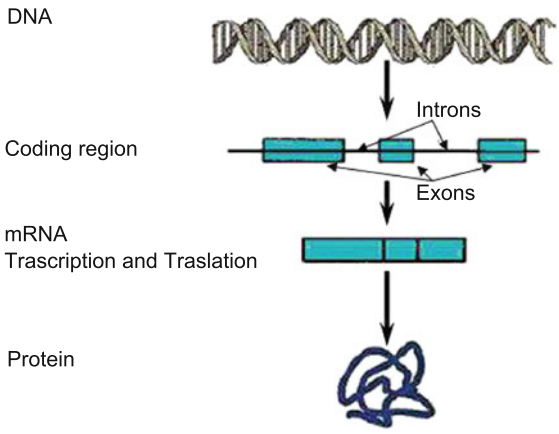

Fig. 7.4 The structure of a coding region and gene transcription

establish that microRNAs play a fundamental role in tumor development, also put forward a research plan for their use in therapy. It is also worthy of mention the way in which genetic information is transmitted to new cells. Indeed, that mechanism is what indissolubly connects the DNA molecule to inheritance and evolution. The double helix structure of DNA has already been discussed (see again the box "Textual Representation of DNA") and it is also well known that during cell division, only one helix of DNA is "passed on" to a new cell. In that new cell, DNA will appear again in its double helix form thanks to DNA polymerase which reproduces the missing helix from the one that is present. DNA recombination and copying are error-prone processes and therefore variations in a genomic sequence may be introduced. The simplest one is a substitution, consisting of one letter being replaced by another in a given genomic position. If that mutation is transferred to the offspring it enters the "big game of evolution". The human genome is made up of nearly three billion bases, and it is estimated that the difference between any two given genomes is on the order of about three million bases. Nature, which seems to appreciate combinatorics and make extensive use of it, leaves room for $4^{3,000,000}$ different human genomes. Moreover, although it would be nice to think of ourselves as being a "unique" species, 99 % of our genetic code is very similar to that of other mammals. In addition, many of our genes are similar to those of many other species, including fruit flies, worms and ... bacteria. In fact, winning biochemical mechanisms are preserved or, more precisely, the mechanisms that are preserved are the winning ones. For instance, histones are among the most conserved eukaryotic proteins and that is a clear indication of their involvement in fundamental biological processes. A guiding principle that one can abstract is that, even in biodiversity, genomic and proteomic similarity is a notable indication of biological relevance. Such a principle gives rise to one of the fundamental working hypotheses of computational biology:

Similarity of genomic or proteomic sequences and structures, as measured by suitable mathematical functions, is a strong indication of biological relatedness, in evolutionary and/or functional terms.

Equipped now with that working hypothesis, apparently very fragile given the complexity of the "machine of life", we will now enter into some areas where algorithmic research has obtained some very valuable successes.

7.3 Algorithmic Paradigms: Methodological Contributions to the Development of Biology as an Information Science

> . . .The discovery of DNA structure started us on this journey, the end of which will be the grand unification of the biological sciences in the emerging, information-based view of biology.[2]

A genomic sequence contains two types of digital information, suitably represented: (a) the genes encoding the molecular machinery of life, the proteins and the RNA, (b) the interaction and regulatory graphs that specify how these genes are expressed in time, space and intensity. Moreover, there is a "hierarchical flow of information" that goes from the gene to the environment: gene → protein → protein interactions → protein complexes → graphs of protein complexes in the cell → organs and tissue → single organisms → populations → ecosystem. The challenge is to decipher what information is contained within this digital code and in the hierarchical flow that originates from it. Since a few years after the discovery of DNA structure and the first sequencing experiments, algorithmics has played a key role in that decoding process. In order to point out the impact that such algorithmic studies have had on the life sciences, both in terms of tools and methodologies, it suffices to mention the following two examples. The BLAST program, the result of deep studies combining statistics and algorithmics, is a working tool that is now indispensable for the analysis of biological sequences. The sequencing of a genome by the shotgun sequencing techniques is now a consolidated reality, but it had a rather controversial beginning. One of the first studies to clearly indicate the feasibility of that type of sequencing on a genomic scale is based on algorithm theory. In what follows, we briefly present algorithmic paradigms, i.e., general approaches, that have made fundamental contributions in several areas at the forefront of research in the life sciences.

[2] Hood and Galas [61].

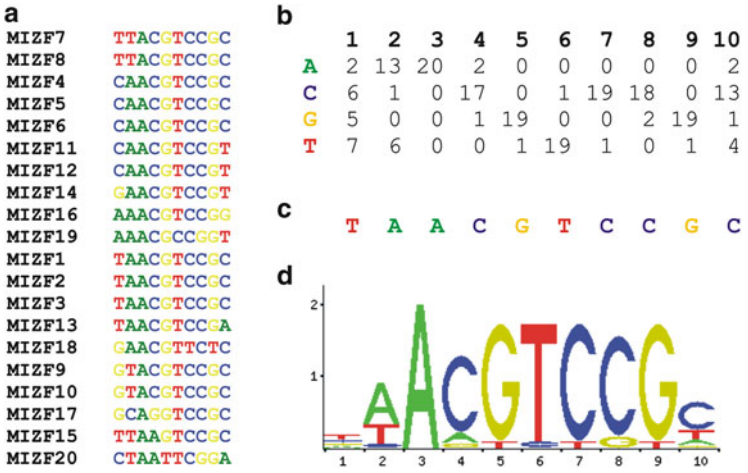

Fig. 7.5 Binding sites and motif representation

7.3.1 String Algorithmics: Identification of Transcription Factors Binding Sites

From the brief introduction given in Sect. 7.2, it is clear that one of the main areas of research in molecular biology is the discovery and understanding of the mechanisms that regulate gene expression, which have a strong implication also for medicine. To this end, an important line of research is the identification of regulatory regions, in particular binding sites of transcription factors. Obviously, carrying out such a discovery process purely via computational methods would be a great result for several reasons, e.g., the cost of experiments in silico compared to those in vitro. Unfortunately, this problem is difficult for many reasons, the main one being that the sequence that corresponds to a specific binding site is very short, 8–10 bases, and its retrieval could encompass the scrutiny of very long sequences, e.g., thousands of bases. It would be like looking for a needle in a haystack.

Transcription factors, in order to do their job, must bind to a particular DNA region. It is quite common that a transcription factor binds to a set of sequences instead of a single specific one. Those sequences are in different parts of the genome and usually share some characteristics that differentiate them from the other sequences in the genome. These common characteristics make it possible to describe these sites by a "motif", which can be defined in different ways. For the convenience of the reader, the notion of motif will be exemplified. Figure 7.5a shows the alignment of 20 sequences of binding sites of the transcription factor MIZF (zinc finger), i.e., a superimposition of the sequences summarized in a table. Figure 7.5b represents a Position Weight Matrix, i.e., a matrix of nucleotide frequencies in the positions of the alignment. Figure 7.5c shows the pure majority system: the motif is obtained as the consensus sequence from the matrix of frequencies electing the

character with a relative majority in each column. Figure 7.5d exemplifies the pure proportional system: the motif is represented as a "logo" of the sequence obtained again from the matrix of frequencies: each character has a height, in the interval [0, 2], proportional to its frequency at that position. In general, given a specific transcription factor, the mining of its binding sites consists of finding all the shared sequence features in the site sequences. The main problem is the vague knowledge of the exact positions of interest in the genome. In particular, this knowledge is usually represented by a set of sequences, each sequence in the set corresponds to one or more binding sites.

The direct approach to the problem of extracting motifs from sets of sequences (each one several hundred bases long) offers only solutions based on enumeration and therefore expensive in terms of time. Fortunately, by taking advantage of statistical information about the relevance of each candidate motif, it is possible to reduce the search space, substantially improving the computational time of the algorithms.

A recent study has evaluated the best-known algorithms in the literature (11 as of 2005), on a benchmark dataset consisting of sequences with known binding sites. The performance of an algorithm is evaluated based on the percentage of binding sites correctly identified. Among the 11 algorithms examined, the best is Weeder, an algorithm that uses, in a very brilliant and original way, data structures and statistical counting techniques representative of many algorithms designed for the mining of textual information. The core of the algorithm is the suffix tree, a ubiquitous data structure used to represent textual information. In the area of data structures, the suffix tree is one of the most fundamental and useful ones: it has been developed from basic research and is largely used in several applications in bioinformatics. (The box "The Suffix Tree Data Structure" shows an example of a suffix tree and details some of its features.) A few years later, the algorithm MOST was developed to solve the same problem. While differing from Weeder, the core of MOST is still a data structure analogous to the suffix tree. The main idea of both algorithms is the identification of portions of a genomic sequence that are "over-represented" or "under-represented", i.e., portions of a sequence that are repeated more frequently or less frequently than expected. In fact, sequences that have an abnormal "statistical behavior" usually also have an important biological function. The computational complexity of the above algorithms depends, strongly, on the particular instance in input. For instance, for Weeder, the computational time can take from a few seconds to several hours.

The Suffix Tree Data Structure

A suffix tree is a data structure designed to represent a sequence of characters, highlighting the suffixes that comprise it. More in detail, a suffix tree for a given sequence S of n characters is a rooted tree (see Chap. 2) with n leaves.

(continued)

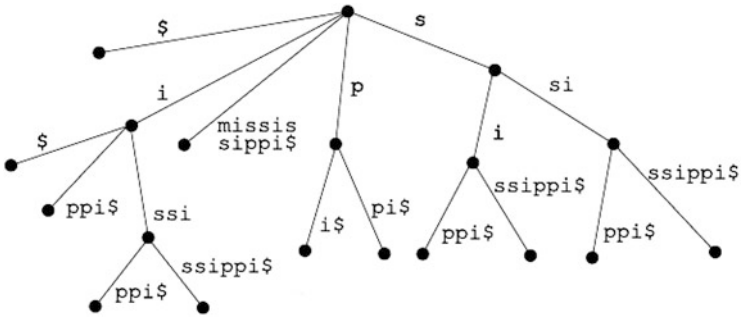

Fig. 7.6 The suffix tree for the sequence mississippi$

(continued)

Each internal node has at least two children and is labeled with a subsequence of S. The concatenation of the labels of the arcs of a path from the root to a leaf determines a specific suffix of the sequence S. To better illustrate this data structure, consider for example the sequence mississippi$. The $ has the formal role of preventing that a suffix is a prefix of another suffix. For example, ignoring the $ symbol in the sequence, the suffix i is the prefix of the suffix *ippi*, while the suffix $i\$$ is not prefixed by any of the suffixes in mississippi$. This property allows for the association, one-to-one, between each suffix and its starting position in the sequence and it is essential for what follows. For example, the suffixes ssissippi$ and ssippi$ correspond to the positions 3 and 6 in the sequence. The suffix tree stores all the suffixes of a given sequence, such that: (a) suffixes with common prefixes share a path from the root to the leaves; (b) for each suffix, there is only one path from the root to a leaf associated with it, and vice versa. Property (b) is a direct consequence of the fact that, by construction, no suffix is a prefix of another in a given sequence. Thanks to the properties (a) and (b), the suffix tree stores all the subsequences of a sequence S and can be used to collect efficiently many statistics about S. Moreover, since suffixes that have prefixes in common share a path from the root of the suffixes tree, one has that identical subsequences will share a path starting from the root. In the suffix tree of Fig. 7.6, the root has five children because there are five different characters in the sequence that we are analyzing (including $). It is also easy to see that every letter appears in the sequence a number of times equal to the number of leaves in the subtree associated to it. Although it may seem surprising, there are algorithms linear in terms of computational complexity (i.e., complexity proportional to the length of the input sequence) able to construct a suffix tree.

7.3.2 Kolmogorov Algorithmic Complexity: Classification of Biological Sequences and Structures

The study of the evolution and classification of species has shifted from the consideration of morphological traits to the consideration of genomic ones. This change of approach has led, for example, to the discovery that the most common laboratory animal is not a rodent, even if it looks like it. Although there is a vast literature in the field and hundreds of algorithms have been designed for evolutionary studies, unfortunately the time performance of most of them does not scale well when it is required to classify very long sequences of bases or whole genomes, instead of sequences consisting of a few thousand bases. Below, we describe a solution to this problem which is particularly elegant, deep, and also effective. To this end, we need to introduce a classic notion of complexity that provides a very elegant measure of the "complexity" of an object, encoded by a text sequence x. This measure, $K(x)$, defined independently by Chaitin[3] and Kolmogorov,[4] and known as Kolmogorov complexity, is given by the length of the shortest program that produces x, without any input. Intuitively, the length of the shortest program required for the automatic construction of x provides a quantification of how complex x is. This insight and the corresponding definition can be extended in several ways. For example, $K(x|y)$ denotes the complexity of describing x (if y is known) and $K(x, y)$ denotes the complexity of describing both x and y. Kolmogorov complexity has several applications in a myriad of contexts, thanks to its links with statistics and the classic information theory founded by Shannon.[5] It is also a very elegant and simple way to quantify how two objects are "similar". In fact, if x is related to y, it is expected that $K(x|y)$ is smaller than $K(x)$. That is, if x and y are similar, the way to describe x starting from y is more concise than that of describing x, starting from nothing. Based on this observation, the theory of Universal Similarity Measures between two sequences has been developed and applied to the classification of sequences and biological structures, providing evidence of the validity of the approach. Further studies have clearly shown that the related biological tools based on this notion are extremely fast, scalable with the amount of data, and flexible. Therefore, they are very competitive with respect to the other previously known methods. For example, given a set of genomes it could be useful to group them in a hierarchical tree, depending on how similar they are based on their sequences. Since the similarity of genomes at the sequence

[3] Gregory John Chaitin is a well-known mathematician. When he came up with the idea and corresponding research on Algorithmic Information Theory he was only 18 years old and had just graduated from CUNY (City College, New York).

[4] Andrey Nikolaevich Kolmogorov was one of the greatest mathematicians of the twentieth century and perhaps that is the reason why this complexity measure carries his name.

[5] Claude Shannon is the founder of a mathematical theory of communication that has taken the name of Information Theory. This theory was born after World War II for a project regarding telecommunications networks, but it has had a very wide set of applications also in other fields.

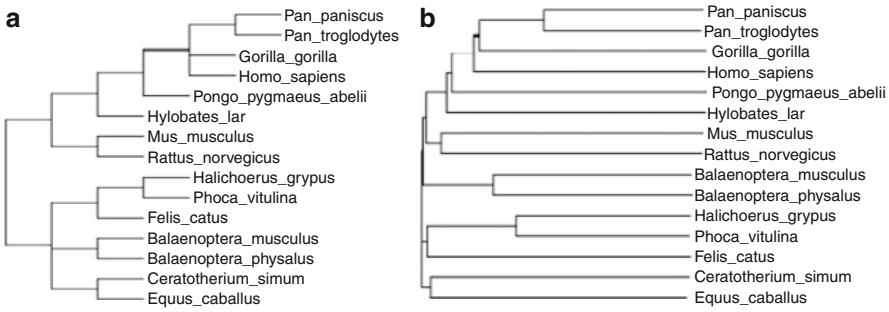

Fig. 7.7 The tree in (**a**) shows the taxonomy of the National Center for Biotechnology Information (NCBI), obtained based on biological considerations and found correct; the tree in (**b**) shows the classification of the same species obtained by using Kolmogorov complexity on mitochondrial DNA sequences

level is indicative of common evolutionary histories, it is expected that the tree produced should be a biologically acceptable phylogeny. To construct such a tree it is necessary to have algorithms that compute the similarity between genomes and then a rule that puts together the genomes, using the quantification of their similarities. This latter information is stored in a matrix, referred to as a similarity matrix, and its calculation is the most expensive step of the entire procedure. The algorithms that calculate similarities via Kolmogorov Complexity are considerably fast and accurate.

To illustrate this point, we consider 16 mitochondrial genomes, whose evolutionary classification is known and validated. The tree in Fig. 7.7a shows the taxonomy of the National Center for Biotechnology Information (NCBI), obtained based on biological considerations; the tree in Fig. 7.7b shows the classification of the same species obtained by applying the computational tools coming from Kolmogorov Complexity to the mitochondrial DNA sequences. The two trees are almost identical, not only "by eye", but also according to a formal mathematical similarity function between trees (the Robinson and Fould distance). In fact, the only difference is that the group of whales in the two trees do not have the same "close relatives". The construction of the second tree took a few seconds on a personal computer, while the first one was obtained using a semi-automatic procedure, involving also expert NCBI biologists who used knowledge available in the literature. Thus, not only is the automatic method fast, but it also provides a good starting point for the biologist to obtain a biologically valid classification.

7.3.3 Graph Algorithmics I: Microarrays and Gene Expression Analysis

In the introduction of this chapter, microarrays were mentioned as a technology that allows for the study of the level of expression of many genes, subject to the same

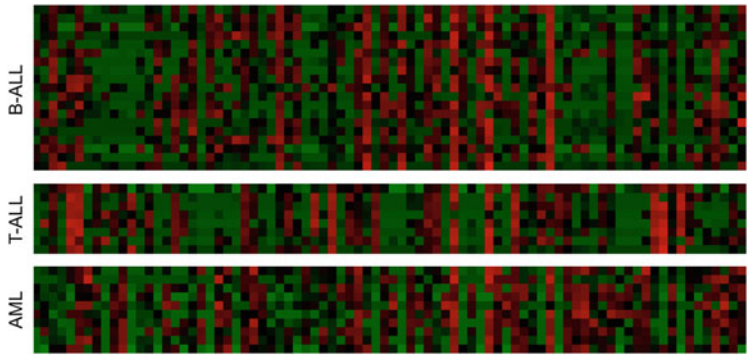

Fig. 7.8 The leukemia microarray for 38 patients (*rows*) and 100 genes (*columns*)

experimental conditions. This procedure gives very useful information in order to determine which genes have similar behaviors (expressed or non-expressed) under similar experimental conditions. The end result of an experiment using microarrays is a numerical matrix, referred to as an expression matrix. Generally, this matrix has a number of rows equal to the number of genes involved in the experiment and a number of columns equal to the number of experimental conditions. When microarrays are used for the molecular classification of tumors, the rows are associated with patients while the columns are associated with genes. The numerical value in the entry (i, j) quantifies the expression levels of gene j in a cell of patient i. A row gives a signature of how the genes behave in the same patient. It is expected that patients with similar diseases have "similar" rows in the expression matrix. The "clustering" of the patients depending on the similarity of the behavior of some genes could lead to the identification of more accurate methods for disease classification.

A brief example may help to clarify those ideas. Figure 7.8 shows, graphically, the gene expression matrix of a leukemia study. The rows are the histological samples of 38 patients and the columns are expression levels of 100 genes, carefully selected from over 16,000 that have been studied for that pathology. The patients are divided into three classes, depending on the type of leukemia affecting them: AML and ALL, this latter being further divided into two groups, the lines T-ALL and B-ALL. Given in graphical form, the different tones of the image correspond to different expression levels. The microarray shown in Fig. 7.8 is part of a study about the classification of tumors on a molecular basis. This analysis led to the construction of computer procedures that are able to accurately diagnose the type of leukemia without the intervention of experts. Therefore, it has produced an automatic diagnostic tool of great help to doctors, particularly those who have no significant experience with that particular disease. This study is the first establishing that it is possible to build diagnostically accurate and clinically relevant tools for the classification of tumors, using microarrays and experimental computational techniques.

From the computer science point of view, one of the main fundamental problems for the analysis of microarray data is the "clustering", i.e., the division of the rows of the expression matrix into "similar" groups, referred to as clusters. Although clustering is a widely studied problem, data from microarrays are very difficult to analyze in this context. This is due to the fact that the rows of the expression matrix are considered as vectors in a high-dimensional geometric space, making this problem very difficult. This situation has revived interest in the development of clustering algorithms, specific for gene expression data. Two of the most recent and distinguished clustering algorithms are Cast and Click. They have an important role in various research topics and are provided in all major analysis platforms for microarray data available in the literature. Following the literature, both algorithms consider the problem of clustering as one of partitioning[6] a given set into disjoint subsets. Formally, the set to be partitioned is a graph that has a vertex for each object to be classified. Then, for each pair of objects (i, j), there is an edge labeled with a certain weight, i.e., a real number which measures the "similarity" between objects i and j, as given by the corresponding rows of the expression matrix. The two algorithms use different techniques for partitioning the graph into subgraphs. Click tries to get the subgraphs formalizing clusters as a problem of network flow, where the edges with a low weight are removed. The theoretical version of Cast tries to build subgraphs as close as possible to complete subgraphs, or subgraphs that do not miss any of the possible edges. Starting from the purely theoretical Cast version, it has been possible to derive a powerful heuristic process for clustering gene expression data, which has the same name and is shown in the box "The Main Steps of Cast".

The Main Steps of Cast

Let s be a similarity function between two objects x and y, with values in [0, 1]—the greater the value of $s(x, y)$, the more similar the objects are; let $S(x, C) = \sum_{y \in C} s(x, y)$ be the total similarity between an element x and a set of elements C, where α is a discrimination parameter with values in [0, 1]. Cast identifies a cluster at a time via an iterative process. Assume that the following "status" holds in a generic iteration of the algorithm: there are elements in the list *UC* that have not been clustered yet and a partial cluster *Ctemp* is under construction. *Ctemp* is modified by two basic steps: ADD and REMOVE. They are performed in the order given and repeated until one can no longer add or remove items from *Ctemp*. At this point, *Ctemp* is declared stable and labeled as a cluster itself. The procedure resumes with

(continued)

[6]The term partition refers to a decomposition of a set of "items" into disjoint subsets, whose union is equal to the entire set.

(continued)

Ctemp empty, in the case that there are still elements to be clustered (*UC* is not empty). In the following, the ADD and REMOVE steps are detailed.

ADD: an element x is chosen from *UC* such that $S(x, Ctemp) \geq \alpha|Ctemp|$ is maximized. That is, the similarity average of x with elements in *Ctemp* must be at least α percent and maximal. If that condition is satisfied, x is included in *Ctemp*, and removed from *UC*.

REMOVE: an element y is chosen from *Ctemp* such that $S(y, Ctemp) < \alpha|Ctemp|$ is minimized. This means that the average similarity of y with elements in *Ctemp* is below α percent and minimal. If that condition is satisfied, y is included in *UC* and removed from *Ctemp*.

Click and Cast are extremely fast and take a few seconds on microarrays with hundred of thousands of genes and conditions, usually providing the number of groups in which it is reasonably possible to divide the set of genes. Finally, we briefly come back to the leukemia data. Experimentally, it has been verified that, given an expression matrix and an appropriate choice of the input parameters, Cast is able to reconstruct the classification in Fig. 7.8, with only two exceptions. It is worth pointing out that such a dataset is "simple" to cluster while, in other circumstances, one must be prepared to have a far lower percentage of accuracy.

7.3.4 Graph Algorithmics II: From Single Components Towards System Biology

Several relevant studies indicate that the identification of interactions between different components, such as proteins, at the level both of single and different organisms, plays a fundamental role in biology. In the following, we briefly highlight that such interactions are of several types and all of them are very important for the identification of cellular machinery and evolution histories that characterize and differentiate species. For example, the human being and the chimpanzee are very similar, both in terms of genome sequences and gene expression levels. However, the interactions between genes (graphs of genes) are very different in the two species, in particular regarding the central nervous system. Like modern electronic devices, many components in humans and chimpanzees are similar, but the difference is in the "circuitry" which determines how these components interact. Discovering the similarities and differences of this circuitry, among various species, provides important information for system biology, where one of the main goals is the identification and the understanding of the fundamental properties of those interactions at the biomolecular "system" level. The significant amount of biological

data produced in the last few years has uncovered several biomolecular interactions that can be encoded as graphs, both for the human being as well as for other model species. In the past 5 years, the number of interaction graphs available has increased by more than one order of magnitude. Moreover, technological advances will allow an exponential growth in the number of such graphs. Such a growth of the available interaction data is similar to that seen in the 1990s for genomic and proteomic sequences. The main difference is that the analysis of genomic and proteomic sequences had at its foundation over 40 years of research in algorithmic theory, while there is no such a background for interaction graphs.

In the literature, there are several interaction graphs of biological interest: protein–protein, gene regulation, co-expression and metabolic. Each of those graphs has its own features, but all have in common two computational problems whose solution, when mature, will provide research tools to the life sciences as important as the BLAST tool. The first problem is to identify, given a set of graphs, the common structure shared by them. That is, subgraphs that appear similar in all graphs. This problem is very similar to, but much more difficult than, the identification of patterns in sequences (see Sect. 7.3.1). In fact, it can be phrased again as a motif discovery problem, but this time in terms of graph structures. The other problem is: given a "query" graph, identify all graphs in a database in which there appear subgraphs similar to the query graph. Although there is a characterization of the mentioned two problems in terms of computational complexity, their study for the design of algorithms for biological graphs is in its infancy.

In what follows, we provide a short description of NetMatch, an algorithm that, given a "text" graph and a "query" graph, identifies in the text graph the subgraphs similar to the query graph according to a precise notion of similarity.

For example, consider the protein–protein interaction graph of yeast (*Saccharomyces cerevisiae*), annotated with gene ontology, shown in Fig. 7.9a. Assume one wants to find, in that graph, paths starting from proteins localized in the plasma membrane to proteins in the nucleus, passing through kinase proteins. This request is encoded by the query graph shown in Fig. 7.9b. Note that the nodes in the graph in Fig. 7.9b are connected by dashed edges. This encodes the fact that these edges can be replaced by paths during the search process. The graph in Fig. 7.9c represents an answer to the query. From the algorithmic point of view, the problem addressed by NetMatch (isomorphism between subgraphs) is computationally difficult, i.e., it is an NP-complete problem and is conjectured to have an exponential time complexity (see Chap. 3). Since the isomorphism between subgraphs is a fundamental problem in many contexts, different heuristic solutions have been studied. NetMatch generalizes to the case of "approximate isomorphisms" some of the known algorithms in the literature for the exact solution. In particular, in order to have a fast program on biomolecular graphs, several engineering speed-ups have been used. In fact, the program is able to perform in a few seconds approximate searches on complex interaction graphs.

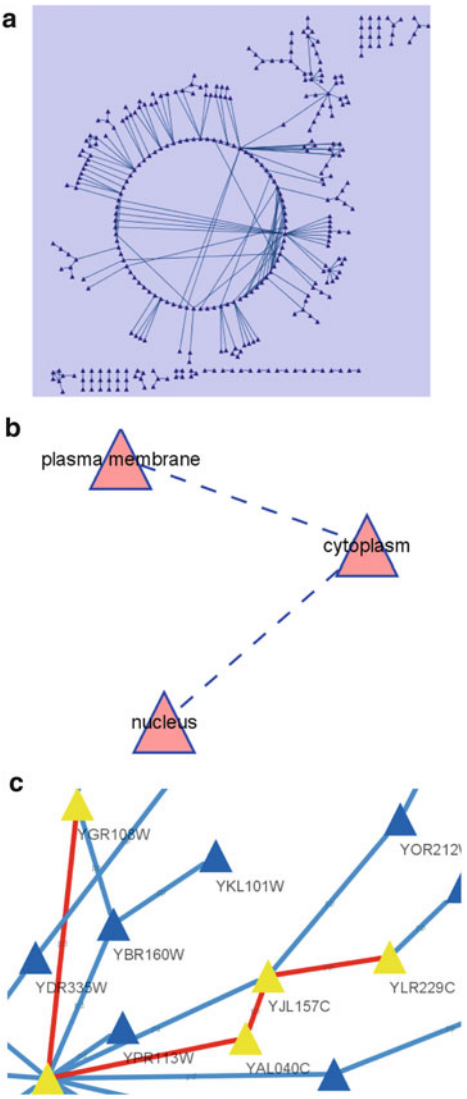

Fig. 7.9 An example of protein interaction search in a biomolecular circuit. (**a**) A protein–protein interaction graph; (**b**) the "query" graph; and (**c**) result of the query

7.4 Future Challenges: The Fundamental Laws of Biology as an Information Science

The paradigm shift, just started, leading to the extension of the information sciences to biology raises important questions that will affect both the biological and the information sciences. It is quite clear that this new branch of the information sciences has a multidisciplinary nature, and biologists, physicists, chemists, computer scientists and mathematicians will play a key role. It is also clear that from

this new process a new science and technology will arise, and we have seen only the beginning of it. Such a process requires a huge cultural change, even in the way of teaching all the involved disciplines to new students wishing to contribute to the development of this new branch of the information sciences. On this basis, the classic information sciences have to solve a first great challenge: the characterization of the complex biological information in terms of mathematics and computer science. That is, given the stochastic nature of biological processes we want to discriminate the "relevant biological signal" from the "noise introduced by stochastic processes" in biological data. A brief overview of past efforts could help us to understand what we need in the future. Turing and Shannon, between the 1930s and the 1940s, developed theories that reveal some of the basic laws for the transmission and processing of information, which have led to the development of the foundations that are the basis of the modern "information society" (see Chap. 5). The revolutionary contribution made by those two scientists was to show that something as impalpable as information could be defined, quantified and processed with mathematical tools. Over the years, those theories have become more refined and sophisticated. Moreover, they have led to the development of practical tools used in the transmission of signals and data processing. To all of this, algorithmics has provided vital contributions not only by introducing paradigms but also by proving that there are some intrinsic limitations on how efficiently a problem can be solved by a computer. However, despite the already great knowledge the information sciences have, it does not seem sufficient to provide adequate tools for the characterization and interpretation of "biological complexity". Indeed, the issue of *Science* dedicated to the human genome sequence mentions in its conclusions about ten notions of mathematical complexity known in the literature, but none of them seems to be suitable to characterize "real" biological complexity. The definition and use of this new notion of complexity seems to be a main priority for an information-based approach to biology. Algorithmics for the life sciences can only take advantage from such a foundational support in order to establish the complexity and the amount of information contained in a biological system. On the other hand, it can contribute to the development of this new notion by providing increasingly more accurate research tools to identify biologically meaningful events in the current information overflow characterizing the life sciences.

7.5 Bibliographic Notes

The impact that the discovery of the double-helical structure of DNA has had on science and culture is well presented in a *Nature* special issue that celebrates the 50 years from the discovery of DNA [84]. The essay by Lander [70], although a bit dated, presents the challenges for post-human-genome genomics that are still current and also proposes a global view of biology. Algorithmics has given outstanding contributions to genomic large-scale sequencing, including the human genome, that have not been presented here. Those aspects are presented in [48].

The importance of chromatin in controlling gene expression is well presented in a classic paper by Felsenfeld and Groudine [38], while the involvement of miRNA in cancer was addressed by Calin and Croce [14]. The paper by Hood and Galas [61] is part of the already-mentioned *Nature* special issue. The strong evidence that the guinea pig is not a rodent is presented in D'Erchia et al. [27]. In regard to introductory textbooks presenting algorithms for bioinformatics, those by Gusfield [53] and Jones and Pevzner [63] are worthy of mention. The BLAST sequence alignment algorithm is described in both books. The importance of the suffix tree in bioinformatics is presented in [54]. An elementary presentation of motifs in DNA sequences is given in [29]. Weeder and MOST are presented in [90] and [91], respectively. Two of the papers that develop the research line of classification through Kolmogorov complexity are [40, 73]. More generally, Kolmogorov complexity theory and its applications are well presented in [72]. The study about the leukemia data and the subsequent development of automatic classifiers for such a pathology is presented in [51], while the Click and Cast algorithms are presented in [99], together with many issues regarding microarray data analysis. Fast algorithms for internal validation measures as they apply to microarray data analysis are described in [49]. The state of the art regarding biomolecular graphs is presented in [100, 115]. Finally, NetMatch is described in [41].

Acknowledgements The author is deeply indebted to Luca Pinello and Filippo Utro for helpful discussions and comments about the content of this chapter. Many thanks also to Margaret Gagie for the usual, very competent proofreading and stylistic comments.