© INSTITUT PASTEUR/ELSEVIER
Paris 1991

Res. Microbiol.
1991, 142, 913-916

From data banks to data bases

A. Danchin [1], C. Médigue [2], O. Gascuel [3], H. Soldano [2] and A. Hénaut ([4])

[1] *Unité Régulation de l'Expression Génétique, Institut Pasteur, 75724 Paris Cedex 15,*
[2] *Atelier de BioInformatique, Institut Curie, Section de Physique et Chimie, 75231 Paris Cedex 05,*
[3] *Centre de Recherche en Informatique de Montpellier, 860 rue de Priest, 34100 Montpellier (France), and*
[4] *Centre de Génétique Moléculaire, CNRS, 91190 Gif sur Yvette (France)*

SUMMARY

The information collected in national and international libraries on nucleotide and protein sequences cannot be directly treated for proper handling by existing software. Therefore we evaluated the feasibility of constructing a data base for *Escherichia coli* using the data present in the banks. The knowhow thus acquired was applied to *Bacillus subtilis*. Specific examples of the general procedure are given.

Key-words: Data bank, Data base, Library; *B. subtilis, E. coli,* Nucleotide and protein sequences, Updating, Learning techniques, *In silico*.

Introduction

Accumulation of experimental data is a major challenge to information management in biology. In particular, research programs involving sequencing are faced with geometrically increasing amounts of new data that cannot be handled easily by a scientist. For this reason, several private or public organizations decided to collect some of this information in the form of items with specific descriptive fields in data libraries or banks. The purpose of these collections is to provide information that can be automatically handled, in a form that is as similar to the original information as possible; the authors place the information they wish to communicate directly in the banks.

This form of communicating experimental results is of great use, but there are, however, several severe drawbacks which have to be considered before using it as a base for further development. In particular, in spite of constraints placed on standardization, there are still a wide variety of ways to treat information that is not directly related to the sequence itself (*e.g.* keywords, features, comments, etc.). Even the sequences cannot be entered absolutely without errors or inaccuracies, *e.g.* sometimes authors omit their latest corrections, leave out segments of cloning linkers, or simply give a sequence that is not 100 % exact. This means that in order to extract as much information as possible from a bank, it has first to be transformed into a new information structure that is capable of evolving. In particular, the structure should evolve automatically as new information is incorporated into it from different places; at each new input, the whole base should update its content.

As a preliminary step in this far-reaching goal, we tried to organize data bases for *Escherichia coli* and *Bacillus subtilis* DNA sequences, using available data banks together with sequences supplied by scientists interested in the program. The ultimate goal was to incorporate learning techniques into the procedure so that it would run through the bases and automatically modify the appropriate fields.

When sequencing complete genomes, computers are needed for 3 distinct but interrelated operations: data acquisition, data exploitation and data management. Data bases are specifically related to the last two operations. We shall now describe how we tried to integrate the software into a "spontaneously" evolving data base for *E. coli* and *B. subtilis* sequences.

Acquisition

Needless to say, it is necessary to control the first steps of the process when generating new sequences in order to be able to use the information contained in the whole genome. At present, there are only two sequencing methods: sequencing machines that "automatically" generate sequences from appropriately prepared templates, and the more conventional autoradiography (or any other process generating photons) of gels, followed by reading of the films obtained.

In the former case, software is supplied by the companies that design the machines, and it is virtually impossible to incorporate "raw" data, therefore analysis of systematic errors is difficult. Moreover, it is not possible at present to improve the reading software by using learning techniques. The situation is different in the second case, because it is still possible to design software for reading digitized autoradiograms. It is important to be able to couple the reading automatons to a learning system that can both take into account errors that have been previously identified and use the information about the sequence that is read in order to check its consistency (*e.g.* by local analysis of CDS). Ultimately, it is hoped that analysis of errors may tell us something about the biology of the sequenced DNA.

For software to carry out automatic construction of contigs from individual templates, it must be coupled to data acquisition. This would generate a sequence with normally a very low amount of error (typically less than 1/1,000), which could be further exploited as a source of biological information.

Exploitation

In general, genes are sequenced after cloning and complementation of mutations. The goal of scientists involved in sequencing work has been to identify the gene product (proteins, RNA, DNA regulatory sequences, *etc.*), in order to associate them formally with their genetic and physiological properties. The most common operation is therefore identification of a coding sequence, automatic translation into a polypeptide sequence according to a given genetic code, and comparison of the latter with known sequences present in data banks. For this reason, many software "packages" permit standard treatment of a sequence: identification of open reading frames, translation, comparison with data banks, hydropathy profile of the proteins, secondary structure predictions, etc. Most often, once a gene has been thus studied, not much more is done to it using computers. Clearly this indicates that a large proportion of the information present in the sequence is not exploited. Indeed, a gene is not simply an instantiation of a random set of nucleotides, but it reflects the history of past constraints. This means that many kinds of relationships between genes or gene fragments exist. One is faced, therefore, with the problem of building up knowledge of sequences by induction, a typical feature of learning techniques.

Experimentation *in silico*

Using the data available in libraries on simple organisms such as *Saccharomyces cerevisiae* or *E. coli*, we build up first sketches.

We focussed first on *E. coli* and built up a data bank that would be the "mother" set of data enabling us to construct an evolving data base. By the end of October 1990, more than 1,350 kb of non-duplicated DNA of the *E. coli* genome had already been included in the base, derived by extracting overlaps from the bank (Médigue *et al.*, 1990a,b). This allowed us to make a comparison with the physical map constructed by Kohara and colleagues (Kohara *et al.*, 1987), and with the genetic map derived by Bachmann (Bachmann, 1990).

Although *E. coli* DNA sequences have been obtained from a variety of different laboratories (and from different strains), we found that they fit extremely well the map derived from a single strain, W3110. This demonstrated that polymorphism at the nucleotide level is very low and justified an approach to *E. coli* chromosome analysis by "patchwork" sequencing.

Two sets of experiments were performed on computers (experiments *in silico*) using the consistency of the data extracted.

The first problem was raised by studying signal peptides. It is generally assumed that secretion processes are very similar in bacteria and in eucaryotes, and that they use signal peptides composed of 20 to 30 amino acids with a positively charged amino terminus and a hydrophobic core. In spite of the apparent similarity between eucaryotes and procaryotes in this respect, it is also well known, mainly after studies for industrial processes, that secretion of eucaryotic proteins in bacteria is usually very poor. It was therefore of interest to investigate whether differences between eucaryotic (human) and

CDS = coding sequence.

procaryotic (*E. coli*) signal peptide sequences could be identified. Since the sample sets were small (less than 30 peptides in each set when this study was initiated), standard data analysis was not possible. We therefore decided to use the learning techniques recently introduced in artificial intelligence, in particular, a technique meant to discriminate between samples.

The approach taken was to use a set of elementary descriptors (nature of a residue, position of a residue in the sequence, kinship between residues, etc.) and a grammar generating complex descriptors (*e.g.* aromatic residues are preferentially located towards the carboxyterminus of the peptide). Complex descriptors were tested for discrimination between samples using standard statistical tests (Gascuel and Danchin, 1988). At the end, redundant descriptors were grouped and a single member of each group was retained.

It is worth noting that this method has necessarily a (trivial) solution, *i.e.* a description of every sequence in each of the two sets. Obviously, however, the trivial solution is not of interest, only non-trivial descriptors are worth considering.

In the present case, we generated 17 descriptors that mainly identified an *E. coli* signal peptide (Gascuel and Danchin, 1986). Among those descriptors, only three had previously been identified, and they had not been perceived as discriminating between eucaryotic and procaryotic signals. From our work, it was predicted that discrimination inside eucaryotic signals would be poor. This could be due to the fact that the *Homo sapiens* sample was composed of several classes of signal peptide, corresponding to different secretion processes. It was also evident that the specific features of signal peptides in eucaryotes are variable: work by D. Botstein and his group on *S. cerevisiae* signals fits well with this latter prediction (Kaiser *et al.*, 1987). Further validation of the approach should be performed by the construction of artificial signals and a study of secretion in practice. It should be stressed here that a descriptor is *not* a consensus sequence; a signal peptide can be derived from the association of several *mutually exclusive* descriptors, *e.g.* "if X is present at position i, then Y is present at position j", and "ABC is present at position i, i+1, i+2, with A=X", can both be descriptors.

A second learning approach was tested on data generated by the building up of an *E. coli* data base. For this construction, we made a compilation of sequences present in data libraries and extracted overlaps. We then compared the sequences with the restriction map generated by Kohara *et al.* (1987), on strain W3110. It was thus found that two restriction enzymes, among the set of eight used by these authors, yielded partially erroneous maps. Many *Pvu*II and *Eco*RV sites were present in the sequences that did not appear in the map. Since this was not true for other enzymes (including the most frequent cutter *Bgl*I), and as these were *missing* sites rather than *extra* sites, the most likely explanation is that the method used by Kohara (partial digestion) resulted in a defect in restriction by *Pvu*II and *Eco*RV under certain conditions. We therefore explored the possibily of missing sites in the context of the sites present in the map using learning software, CALM (Soldano and Moisy, 1985). *Pvu*II's behaviour was not unexpected: due to the relatively high frequency of G+C bases surrounding the site, restriction was prevented. However, *Eco*RV behaved differently, suggesting that the enzyme is sensitive to the presence of nucleotides upstream from the site (position −9, −6 and −3) (Médigue *et al.*, 1990b).

It is worth noting that construction of a data base resulted in experimental predictions for the enzymology of a restriction enzyme. This demonstrates that learning techniques can be of use in the constitution of a data base.

Management

The *E. coli* data base, meant to act as a model for building up *B. subtilis* sequences, was compiled with "Macintosh II" hardware, using "4-Dimension" as the general data processing software. Procedures have been built in to automatically extract sequences form EMBL CD-ROM as well as directly from experimentalists. Each sequence record has been split into two linked bases, one for the sequence proper (and its comments), one for bibliographical references. A procedure has been constructed for checking possible mistakes in the assignment of CDS, and for the elimination of duplicates when generating single contigs. Obviously this requires verification from original publications in many instances and cannot be performed automatically. From this, a set of records (each corresponding to a contig) can be generated which define sequences present only once. Appropriate fields are created in order to incorporate difference, when duplicates are not completely identical (this means that a choice must be made which requires biological expertise and which cannot be done automatically either). This will constitute the evolving section of the data base, the fields being automatically modified by the appropriate procedures when the records are updated. A secondary data base, linked to the preceding one, is generated by creating records for the amino acid sequences translated from nucleotide sequences. This has also been conceived to evolve according to procedures that can be started either internally or externally.

As can be seen, we now possess a "clean" set of sequences that can be analysed using clustering techniques. Following the taxinomic approach of Benzecri, work has commenced on a set of 1,000 kbp of non-duplicated *E. coli* DNA (Benzecri, 1982).

References

Bachmann, B. (1990), Linkage map of *Escherichia coli* K-12. *Microbiol. Rev.*, 54, 130-197.

Benzecri, J.P. *et al.* (1982), L'analyse des données. — II. L'analyse des correspondances. Dunod, Paris.

Gascuel, O. & Danchin, A. (1986), Protein export in prokaryotes and eukaryotes: indications of a difference in the mechanism of exportation. *J. mol. Evol.*, 24, 130-142.

Gascuel, O. & Danchin, A. (1988), Data analysis using a learning program, a case study: an application of PLAGE to a biological sequence analysis, *in* "Proceedings of ECAI" (pp. 390-395). Pitman, Munich.

Kaiser, C.A., Preuss, D., Grisafi, P. & Botstein, D. (1987), Many random sequences functionally replace the secretion signal sequence of yeast invertase. *Science*, 235, 312-317.

Kohara, Y., Akiyama, K. & Isono, K. (1987), The physical map of the whole *E. coli* chromosome: application of a new strategy for rapid analysis and sorting of a large genomic library. *Cell*, 50, 495-508.

Médigue, C., Bouché, J.P., Hénaut, A. & Danchin, A. (1990a), Mapping of sequenced genes (700 kbp) on the restriction map of the *Escherichia coli* chromosome. *Mol. Microbiol.*, 4, 169-187.

Médigue, C., Hénaut, A. & Danchin, A. (1990b), *E. coli* molecular genetic map (1,000 kbp): update I. *Mol. Microbiol.*, 4, 1443-1454.

Soldano, H. & Moisy, J.L. (1985), Statistico-syntactic learning techniques. *Biochimie*, 67, 493-498.