SIMULATION OF DNA CUTTING

Francisco José Cisneros, Andrés de la Peña, Cristina Piqueras,
Paula Cordero, Juan Castellanos

Abstract: The simulation of the main molecular operations used in DNA Computing can lead the researchers to develop complex algorithms and methods without the need of working with real DNA strands in-vitro. The purpose of this paper is to present a computer program which simulates a cutting process over DNA molecules which is an essential operation for the DNA computation. This simulation represents a useful tool for a virtual laboratory which is oriented to DNA computations. The results given by the software can show the behavior of a DNA cutting under certain set of restrictive enzymes to carry out the operation in-vitro efficiently.

Keywords: DNA Computing, DNA Simulation, Software Simulation, DNA operations, Bioinformatics.

ACM Classification Keywords: I.6. Simulation and Modelling, B.7.1 Advanced Technologies, J.3 Biology and Genetics

Introduction

DNA Computing is an impressive computer paradigm based on the work made by Leonard M. Adleman [Adleman, 1994], where the first implementation of a computer based on DNA operations solved a hard combinatorial problem using deoxyribonucleic acid molecules. He was able to solve an NP-complete problem using DNA molecules and biological operations. This represented an approach to a massive parallel paradigm.

Molecular computing consists of representing the information of the problem with organic molecules [J.Castellanos, 1998] and to make them react within a test tube in order to solve a problem. The fundamental characteristics of this type of computations are, mainly, the massive parallelism of DNA strands and the Watson-Crick complementarity. The speed of calculation, the small consumption of energy and the big amount of information which DNA strands are able to store are the best advantages that DNA computing has. Nevertheless one of the problems is the massive calculation space needed, which limits the size of the problems.

The nucleic acids are linear polymers in which the repetitive unit is the nucleotide. Each nucleotide is formed by a pentose (the ribose or the deoxyribose), a nitrogenous base (purin or pyrimidin) and a phosphoric acid. The union of the pentose with a base constitutes the nucleoside. The union of this last structure with the phosphoric acid gives us the nucleotide. The union of the nucleotides gives us the polynucleotide. The nitrogenous bases that form each DNA molecule are Adenine (A), Guanine (G), Cytosine (C) and Thymine (T). Those which form each RNA molecule are Adenine (A), Guanine (G), Cytosine (C) and Uracil (U). Double stranded molecules are formed by two strands twisted in a helix. The Adenine of a helix matches the Thymine of the complementary helix by creating two hydrogenate bridges. Also, the Guanine of a helix matches the Cytosine of the complementary three hydrogenate bridges. Therefore, the bases of one strand are united by hydrogenate bridges to the bases of the other strand, forming the base pairs AT and GC.

It is very important to determine which biologic operations could be used for the manipulation of DNA strands. In order to distinguish between the common mathematical operations and the biological procedures which are applied on DNA strands, it is used the term bio-operations to talk about the last ones. Some of the bio-operations that facilitate the manipulation of DNA are the measure of DNA strands, the DNA cutting, the lengthening and shortening of DNA strands, the separation and fusion of DNA sequences (denaturalization and renaturalization) or the ordination by length or electrophoresis among others [2].

In this article it is explained the development of a software that simulates successfully the process of cutting over DNA molecules. The aim of it is to incorporate this cutting tool to a virtual laboratory in which all the operations explained above are implemented. This environment help us to prove how molecules would react to the codifications we develop in-info so that the steps needed in a real laboratory are reduce substantially.

DNA cutting

The DNA cutting is one of the most basic operation in computing with DNA, this is so because allow to manipulate the DNA strands in specific points. The data problem are represented through nucleotides sequences, an, at the same time, this sequences, which represent the problem atomic data, are linked in bigger sequences, representing data sets or lists which normally are associated to possible solutions. In order to carry out the manipulation, such as atomic elements extraction, elimination, combination or addition to the possible solutions, generally is necessary to apply the cutting operation. This cutting operation is carried out applying restriction enzymes, which are in charge of making the operation in a parallel massive manner [Kobayashi, 2001].

Enzymes are biomolecules that catalyze chemical reactions. Restriction enzymes (or restriction endonuclease), found in bacteria and archaea, are enzymes that cuts double-stranded or single stranded DNA at specific recognition nucleotide sequences known as restriction sites [Roberts, 2007]. There are three types of restriction enzyme. Such that are included in type I are characteristic of two different strains of *E. coli*. These enzymes cut at a site that differs, and is some distance (at least 1000 bp) away, from their recognition site. The recognition site is asymmetrical and is composed of two portions – one containing 3-4 nucleotides, and another containing 4-5 nucleotides – separated by a spacer of about 6-8 nucleotides. Type II restriction enzymes [Roberts, 2005] are composed of only one subunit, their recognition sites are usually undivided and palindromic and 4-8 nucleotides in length, and they recognize and cleave DNA at the same site. Type III restriction enzymes recognize two separate non-palindromic sequences that are inversely oriented. They cut DNA about 20-30 base pairs after the recognition site.

Type II enzymes are the most commonly available and used restriction enzymes. They are specially good in computing with DNA due to the atomic data are codify frequently for space reasons, with nucleotides sequences the shortest as possible, often smaller than the distance between the cutting point and the recognition site.

Examples of restriction enzymes include [Roberts, 1980]:

Enzyme	Source	Recognition Sequence	Cut
EcoRI	*Escherichia coli*	5'GAATTC	5'---G AATTC---3'
		3'CTTAAG	3'---CTTAA G---5'
EcoRII	*Escherichia coli*	5'CCWGG	5'--- CCWGG---3'
		3'GGWCC	3'---GGWCC ---5'
SmaI*	*Serratia marcescens*	5'CCCGGG	5'---CCC GGG---3'
		3'GGGCCC	3'---GGG CCC---5'

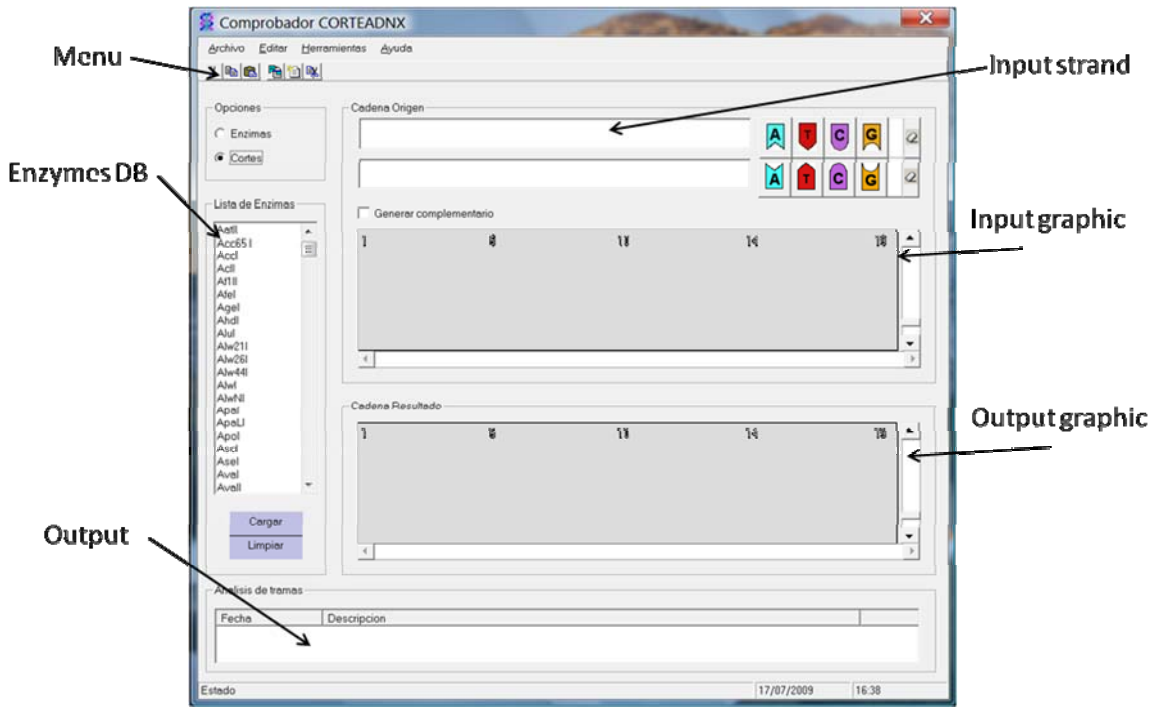
The number of commercial available restriction enzymes is constantly increasing [Roberts, 2003], fact that allow to improve the computation algorithm, since this ones are often limited, especially due to the relatively reduce amount of enzymes with known cut.

DNA cutting simulation

The software of simulation realized allows us establish all recognition sites of DNA strand, and how remain the cuts and the resultants sub chains. The simulator development has being dealt in four parts, the interface, an enzymes database, the cutting simulator and the obtained results viewer, this tow last are ActiveX objects, allowing its use in others simulators, in such a way is possible to combine this ones in a common way. The entire application is written in Visual BASIC 6.

The data access is carried out through and interfaces ODBC, which allow establishing the connection with a wide amount of existing database and in a constant updating process. This is an especially important factor given the permanent emerging of new enzymes allowing, not only enriching languages, but also, in some cases, they contribute with the capacity of carry out operations that improve the algorithm performance. It is attached a database with the 280 more frequently used enzymes.

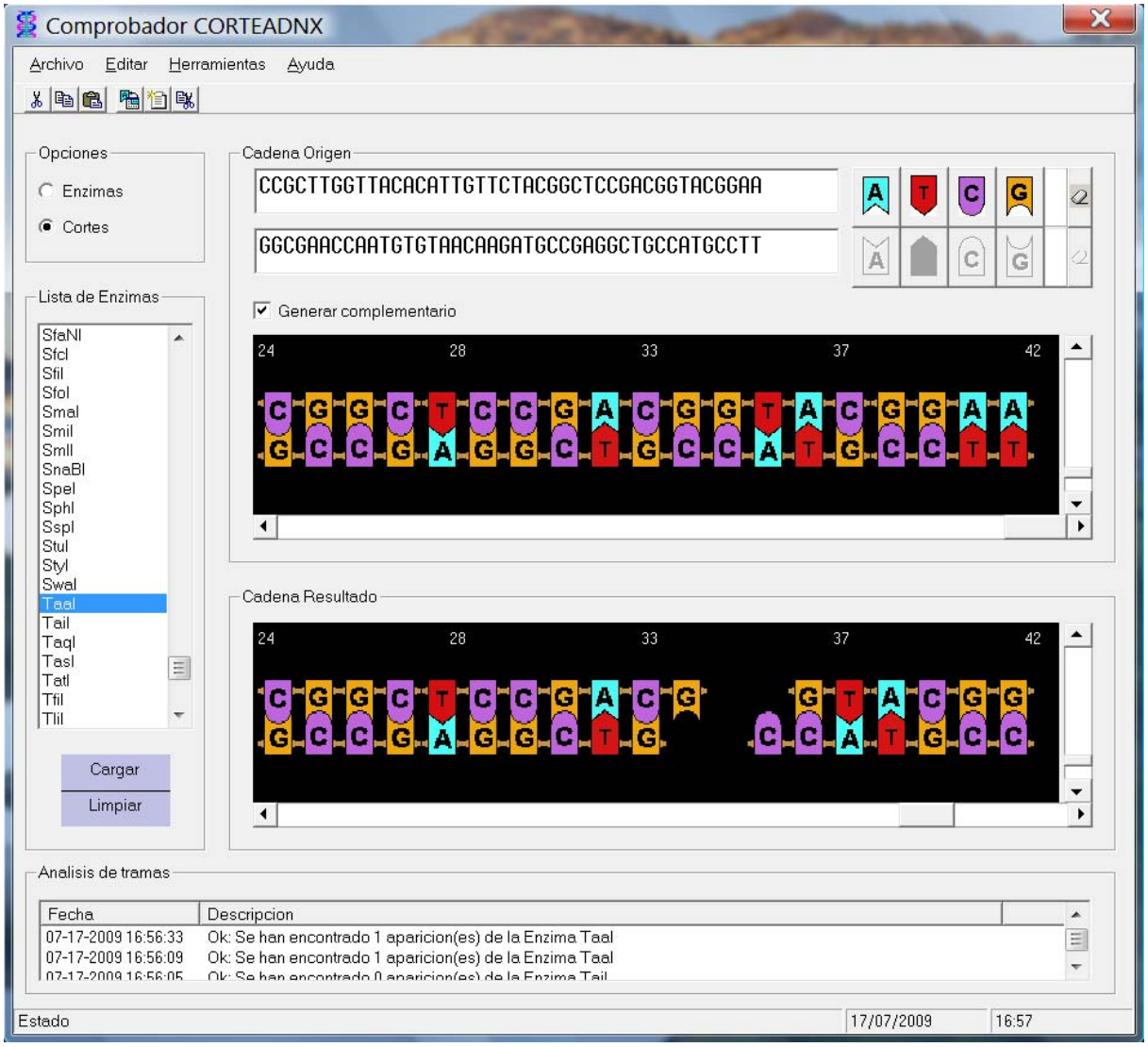
The interface is easy and intuitive, in such a way it could be used without the need of a previous formative course; the X Figure shows the different simulator aspects.



The simulator allow to visualize graphically the application result of an enzyme over a given chain, or, in other way, it is possible to generate a result report of applying every enzymes available over the strand, identifying which find out its recognition site.

Example

Bellow its is presented an example where it is detected the entire amount of enzymes which have a recognition site, content in the chain introduced in order to know the enzymes which could cut the chain. It is a typical example, showing when the anti complementary and stable languages are designed, which need to count with ruptur points, assuring in such a way that no one enzyme will produce a cut in an improper site.



As we can see in the results and reviewing the report that gives us the simulator, we found that the chain can be cut with five enzymes (BspLI, Csp6I, NlaIV, RsaI, TaaI). The chosen screenshot shows the cut made by TaaI enzyme as a checking.

Conclusion

The simulation model here presented allows us to carry out experiments with DNA in a simple way using only our computer. By using it we are able to understand the behavior of DNA molecules in a cutting process without the costs, the time and the space needed in a laboratory. That is why it exist the need of making these experiments easier by simplify the main bio-operations and bio-molecular processes over DNA molecules. It is very important and useful to develop this kind of software simulators so that researchers can create more complicated algorithms for DNA computations without the need of testing in a DNA pool every single operation.

This software try to give a simple view of the behavior of DNA molecules when it is applied a certain cutting enzyme. This shows the exact cutting point and the final shape of each resulting strands, and also is able to recognize the entire amount of enzymes that could cut the chain. The use of this simulator can give a result of a cutting process of the DNA molecules introduced in a faster way than in-vitro.

Bibliography

[Adleman, 1994] Leonard M. Adleman. Molecular Computation of Solutions to Combinatorial Problems. Science (journal) 266 (11): 1021-1024. 1994.

[Adleman, 1998] Leonard M. Adleman. Computing with DNA. Scientific American 279: 54-61. 1998

[Lipton, 1995] Richard J.Lipton. Using DNA to solve NP-Complete Problems. Science, 268:542-545. April 1995

[J.Castellanos, 1998] J.Castellanos, S.Leiva, J.Rodrigo, A. Rodríguez Patón. Molecular computation for genetic algorithms. First International Conference, RSCTC'98.

[Shannon, 1949] C.E.Shannon. The Mathematical theory of communication. In: The Mathematical Theory of Communication. Ed. C.E.Shannon and W.Weaver. University of Illinois Press, Urbana, 1949.

[Roberts, 1980] Roberts RJ. Restriction and modification enzymes and their recognition sequences. Nucleic Acids Res. 8 (1): r63-r80. doi:10.1093/nar/8.1.197-d. PMID 6243774.

[Roberts, 1976]Roberts RJ. Restriction endonucleases. CRC Crit. Rev. Biochem. 4 (2): 123-64. doi:10.3109/10409237609105456. PMID 795607.

[Kessler, 1990] Kessler C, Manta V. Specificity of restriction endonucleases and DNA modification methyltransferases a review (Edition 3). Gene 92 (1-2): 1-248. doi:10.1016/0378-1119(90)90486-B. PMID 2172084.

[Pingoud, 1993] Pingoud A, Alves J, Geiger R. Chapter 8: Restriction Enzymes. in Burrell, Michael. Enzymes of Molecular Biology. Methods of Molecular Biology. 16. Totowa, NJ: Humana Press. pp. 107-200. ISBN 0-89603-234-5.

[Kobayashi, 2001] Kobayashi I. Behavior of restriction-modification systems as selfish mobile elements and their impact on genome evolution. Nucleic Acids Res. 29 (18): 3742-56. doi:10.1093/nar/29.18.3742. PMID 11557807.

[Roberts, 2005] Roberts RJ (April 2005). How restriction enzymes became the workhorses of molecular biology. Proc. Natl. Acad. Sci. U.S.A. 102 (17): 5905-8. doi:10.1073/pnas.0500923102. PMID 15840723. PMC: 1087929. http://www.pnas.org/cgi/pmidlookup?view=long&pmid=15840723.

[Roberts, 2007] Roberts RJ, Vincze T, Posfai J, Macelis D. (2007). REBASE--enzymes and genes for DNA restriction and modification. Nucleic Acids Res 35 (Database issue): D269-70. doi:10.1093/nar/gkl891. PMID 17202163. http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pubmed&pubmedid=17202163.

[Adrianne, 2001] Adrianne Massey; Helen Kreuzer (2001). Recombinant DNA and Biotechnology: A Guide for Students. Washington, D.C: ASM Press. ISBN 1-55581-176-0.

[Pingoud, 2001] Pingoud A, Jeltsch A (September 2001). Structure and function of type II restriction endonucleases. Nucleic Acids Res. 29 (18): 3705-27. doi:10.1093/nar/29.18.3705. PMID 11557805.

[Roberts, 2003] Roberts RJ, Belfort M, Bestor T, Bhagwat AS, Bickle TA, Bitinaite J, Blumenthal RM, Degtyarev SKh, Dryden DT, Dybvig K, Firman K, Gromova ES, Gumport RI, Halford SE, Hattman S, Heitman J, Hornby DP, Janulaitis A, Jeltsch A, Josephsen J, Kiss A, Klaenhammer TR, Kobayashi I, Kong H, Krüger DH, Lacks S, Marinus MG, Miyahara M, Morgan RD, Murray NE, Nagaraja V, Piekarowicz A, Pingoud A, Raleigh E, Rao DN, Reich N, Repin VE, Selker EU, Shaw PC, Stein DC, Stoddard BL, Szybalski W, Trautner TA, Van Etten JL, Vitor JM, Wilson GG, Xu SY. A nomenclature for restriction enzymes, DNA methyltransferases, homing endonucleases and their genes. Nucleic Acids Res. 31 (7): 1805-12. doi:10.1093/nar/gkg274. PMID 12654995.

[Geerlof, 2008] Geerlof A. Cloning using restriction enzymes. European Molecular Biology Laboratory - Hamburg. http://www.embl-hamburg.de/~geerlof/webPP/genetoprotein/cloning_strategy/clo_rest-enzymes.html.

[Wolff, 2008] Wolff JN, Gemmell NJ. Combining allele-specific fluorescent probes and restriction assay in real-time PCR to achieve SNP scoring beyond allele ratios of 1:1000. BioTechniques 44 (2): 193-4, 196, 199. doi:10.2144/000112719. PMID 18330346.

[Zhang, 2005] Zhang R, Zhu Z, Zhu H, Nguyen T, Yao F, Xia K, Liang D, Liu C. "SNP Cutter: a comprehensive tool for SNP PCR-RFLP assay design". Nucleic Acids Res. 33 (Web Server issue): W489-92. doi:10.1093/nar/gki358. PMID 15980518.

Authors' Information

Francisco J. Cisneros – Natural Computing Group, Universidad Politécnica de Madrid, Boadilla del Monte, 28660 Madrid, Spain, e-mail: kikocisneros@gmail.com

Andrés de la Peña – Natural Computing Group, Universidad Politécnica de Madrid, Boadilla del Monte, 28660 Madrid, Spain, e-mail: andres.de.la.penya@gmail.com

Cristina Piqueras – Natural Computing Group, Universidad Politécnica de Madrid, Boadilla del Monte, 28660 Madrid, Spain

Paula Cordero – Natural Computing Group, Universidad Politécnica de Madrid, Boadilla del Monte, 28660 Madrid, Spain, e-mail: paula.cormo@gmail.com

Juan Castellanos – Natural Computing Group, Artificial Intelligence Department, Facultad de Informática, Universidad Politécnica de Madrid, Boadilla del Monte, 28660. Madrid, Spain, e-mail: jcastellanos@fi.upm.es