INTERNATIONAL JOURNAL OF NATURAL AND APPLIED SCIENCES (IJNAS), VOL. 12, Special Edition (2019); P. 61 – 73, 2 TABLES, 6 FIGS.

Diversity in Content and Analytical Algorithms of Biologist-Centric Software(s) for DNA and Sequence Data: Mega, DNASp, GenAlex and ARLEQUIN.

F. U. Ogban[1*], U. O. Udensi[2], G. A. Inyang[1], F Osang[3]

ABSTRACT

The proliferation of analytical software in life science, specifically in DNA and protein sequence data analytics has exposed users and researchers to a dilemma of extensive review of such software in order to ascertain and consider which of the software is most suitable for their application. The relative merits and the associated focus of these different software packages remain unclear. Of essence is the fact that users and researchers are often in the quest for the identification of genomic variants at different levels. Of the much, our attention is on (a) the Molecular Evolutionary Genetics Analysis (MEGA) version 7, (b) DNA Sequence Polymorphism (DNASP) version 5.10, (c) the Genetic Analysis in Excel (GenALEx) version 6.5 and (d) the Integrated Software Package for Population Genetics Data Analysis (ARLEQUIN) version 3.5. Criteria to compare their developmental philosophy, interfaces, algorithms and efficiency were developed. The results presented here would give a user and or a researcher a firsthand direction in their quest for the best application that suits their needs. Our result also provides insights into the need for standardization of life science analytical software modules based on developer/users software quality measure.

INTRODUCTION

Next-generation sequencing techniques are demonstrating promise in transforming research in life sciences. These techniques support many applications including metagenomics (Qin *et al.*, 2010), detection of SNPs and genomic structural variants (Alkan*et al.*, 2009; Medvedev *et al.*, 2009) in a population.

The advent of next-generation sequencing (NGS) techniques presents many novel opportunities for many applications in life sciences. The vast number of short reads produced by these techniques, however, pose significant computational challenges. The first step in many types of genomic analysis is the mapping of short reads to a reference genome, and several groups have developed dedicated algorithms and software packages to perform this function. As the developers of these packages optimize their algorithms with respect to various considerations, the relative merits of different software packages remain unclear. However, for scientists who generate and use NGS data for their specific research projects, an important consideration is choosing the software that is most suitable for their application. Some applications (e.g. metagenomics) require *de novo* sequencing of a sample (Miller *et al.*, 2010), while many others (e.g. variant detection, cancer genomics) require resequencing.

For all of these applications, the vast amount of data produced by sequencing runs poses many computational challenges (Horner *et al.*, 2010). Inyang et al (2019)

Genome sequencing is generating vast amounts of DNA sequence data from a wide range of organisms. As a result, gene sequence databases are growing rapidly. In order to conduct efficient analyses of these data, there is a need for easy-to-use computer programs, containing fast computational algorithms and useful statistical methods.

MEGA 7

The objective of the *MEGA* software has been to provide tools for exploring, discovering, and analyzing DNA and protein sequences from an evolutionary perspective. The first version was developed for the limited computational resources that were available on the average personal computer in early 1990s. *MEGA1* made many methods of evolutionary analysis easily accessible to the scientific community for research and education. *MEGA2* was designed to harness the exponentially greater computing power and a graphical interface of the late 1990's, fulfilling the fast-growing need for more extensive biological sequence analysis and exploration software. It expanded the scope of its predecessor from single gene to genome wide analyses.

*Corresponding author**.** Email:
[1]Department of Computer Science, University of Calabar, Calabar, Nigeria
[2]Department genetics and biotech, University of Calabar, Nigeria.
[3]Department of computer science, national open University of Nigeria

Two versions were developed (2.0 and 2.1), each supporting the analyses of molecular sequence (DNA and protein sequences) and pairwise distance data. Both could specify domains and genes for multi-gene comparative sequence analysis and could create groups of sequences that would facilitate the estimation of within- and among- group diversities and infer the higher-level evolutionary relationships of genes and species. *MEGA2* implemented many methods for the estimation of evolutionary distances, the calculation of molecular sequence and genetic diversities within and among groups, and the inference of phylogenetic trees under minimum evolution and maximum parsimony criteria. It included the bootstrap and the confidence probability tests of reliability of the inferred phylogenies, and the disparity index test for examining the heterogeneity of substitution pattern between lineages.

MEGA4 continues where *MEGA2* left off, emphasizing the integration of sequence acquisition with evolutionary analysis. It contains an array of input data and multiple results explorers for visual representation; the handling and editing of sequence data, sequence alignments, inferred phylogenetic trees; and estimated evolutionary distances. The results explorers allow users to browse, edit, summarize, export, and generate publication-quality captions for their results. *MEGA4* also includes distance matrix and phylogeny explorers as well as advanced graphical modules for the visual representation of input data and output results. These features, which we discuss below, set *MEGA* apart from other comparative sequence analysis programs

As with previous versions, *MEGA5* was specifically designed to reduce the time needed for mundane tasks in data analysis and to provide statistical methods of molecular evolutionary genetic analysis in an easy-to-use computing workbench. While *MEGA5* was distinct from previous versions, we made a special effort to retain the user-friendly interface that researchers have come to identify with *MEGA*. We have simplified the file activation process, where you may select an analysis before needing to open a file.

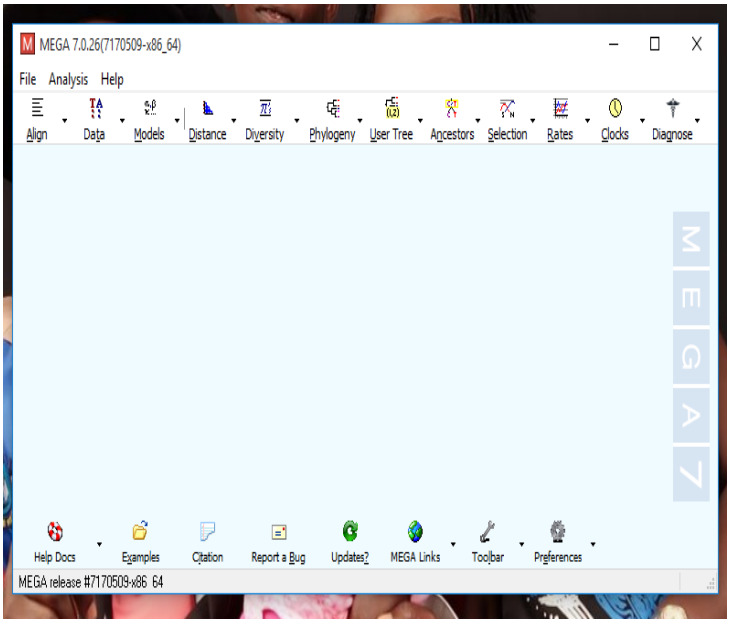

Fig. 1. Interface of MEGA 7.0.26 and its Twelve Menus.

MEGA6 represents a leap forward in terms of performance. The multi-threaded ML system has been optimized for maximum efficiency. A new memory manager and updated compiler have made it possible for *MEGA* to allocate twice as much memory on 64-bit systems as *MEGA5* could. The naïve timing methods that were added in *MEGA5* have been replaced by a RelTime based system which is as accurate as (or better than) contemporary methodologies but with speeds >1000 times faster.

MEGA7 is a major refactoring of the *MEGA* source code and achieves another leap forward in terms of performance. *MEGA* is now optimized for 64-bit processor architectures and can now utilize many GB of memory. In addition, the *Tree Explorer* window has been re-factored to handle trees with > 100k taxa (previously it could only handle ~4k taxa), depending on the available graphics processing resources. Beyond increased performance, improvements have been made to the user interface. The *Time tree* system has been completely restyled to use a wizard system for guiding the user through the steps to create a timetree using the *Reltime* method.

Mega Basics

1. Aligning Sequences
2. Estimating Evolutionary Distances
3. Building Trees from Sequence Data
4. Testing Tree Reliability
5. Working with Genes and Domains
6. Testing for Selection
7. Managing Taxa with Groups
8. Computing Sequence Statistics
9. Building Trees from Distance Data
10. Constructing Likelihood Trees
11. Editing Data Files
12. Constructing Time Trees
13. Inferring Gene Duplications

DNASP 5.10

DnaSP, DNA sequence polymorphism, is an interactive computer program for the analysis of DNA polymorphism from nucleotide sequence data. The program, addressed to molecular population geneticists, calculates several measures of DNA sequence variation within and between populations (with or without the sliding window method) in noncoding, synonymous or nonsynonymous sites; linkage disequilibrium, recombination, gene flow and gene conversion parameters; and some neutrality tests, Fu and Li's, Hudson, Kreitman and Aguadé's, McDonald and Kreitman, and Tajima's tests. DnaSP can also conduct computer simulations based on the coalescent process.

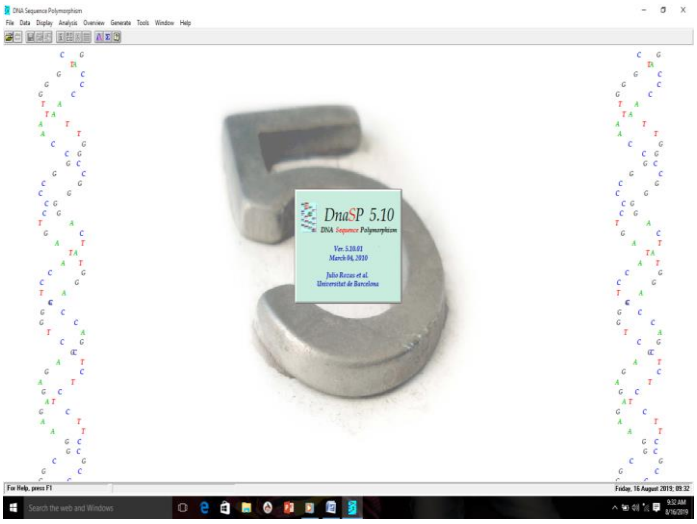

Fig. 2. The interface of DNAsp version 5.10

Version 5 implements a number of new features and analytical methods allowing extensive DNA polymorphism analyses on large data sets. Among other features, the newly implemented methods allow for: 1) analyses on multiple data files; 2) haplotype phasing; 3) analyses on insertion/deletion polymorphism data; 4) visualizing sliding window results integrated with available genome annotations in the UCSC browser.

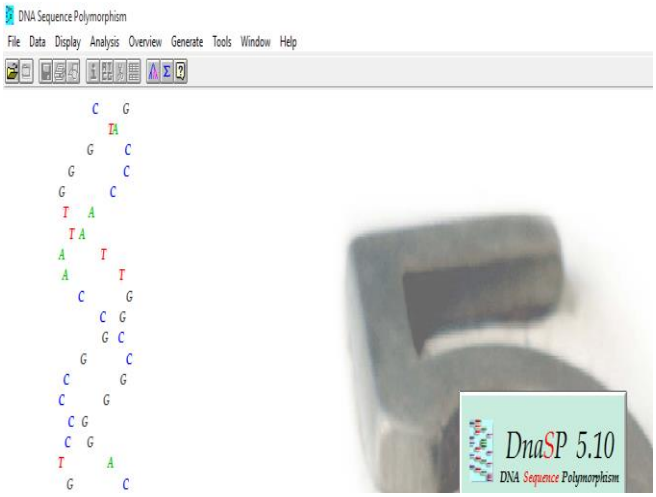

Fig. 3. The nine menus of DnaSP

What DnaSPcannot do:

DnaSP cannot align sequences. There are some available programs that can do this. For example, you can perform the multiple alignments with CLUSTAL-W (Thompson et al. 1994). This program produces an output (multiple aligned sequences in NBRF/PIR format) that can be read by DnaSP.

DnaSP cannot make phylogenetic inferences or manipulate trees. There are many programs to do this: MEGA (Kumar et al. 1994), PHYLIP (Felsenstein 1993), PAUP (Swofford 1991). Nevertheless, the input file formats used by DnaSP (FASTA, MEGA, NBRF/PIR, NEXUS and PHYLIP format) are also recognized for some of them.

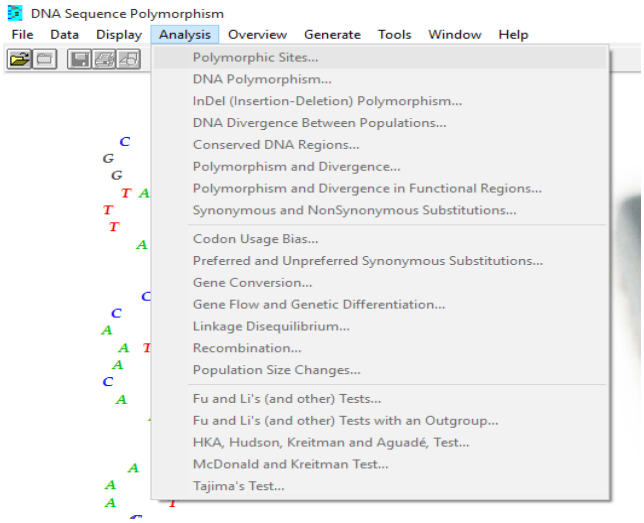

Fig. 4. DnaSP Analysis options

DNA sequences cannot be edited or manipulated by DnaSP. You can do this by using, for example, SeqApp / SeqPup programs.

DnaSP cannot directly analyze diploide genetic information (for instance, SNPs data from diploid genomic regions). If you are using diploid unphase data, you can reconstruct the phase using the Open Unphase/Genotype Data module

GenAlEx

GenAlEx 6.5 - Genetic Analysis in Excel is written in Visual Basic for Applications (VBA) within Excel. It is designed as a user-friendly package that allows users to analyse a wide range of population genetic data within a software environment with which most users will be familiar. However, it is a macro program that must be enabled to run in Excel.

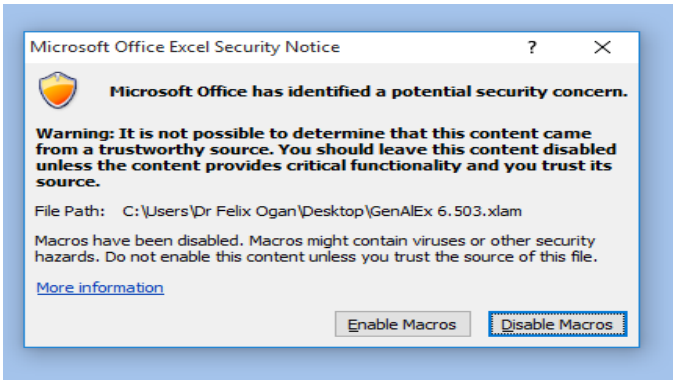

Fig. 5. GenAlEx Macro window for Enabling.

GenAlEx is limited by Excel to 256 columns of data in Excel 2003 (in a .xls workbook) and to 16,384 columns in Excel 2010 (in a .xlsx, .xlsm or .xlsb workbook). This equates to 254 binary or haploid loci or 127 codominant loci in Excel 2003; while, users operating in Excel 2010 are limited to 16,382 binary or haploid loci or 8,191 codominant loci. The maximum number of samples is approximately 65,500 in 2003 and over one million in 2010. In practice, in Excel 2010 onwards, the memory limitations of your computer and the GenAlEx program itself will limit the size of the dataset you can run to far less than the number of columns or rows available. However, analyses have been successfully run for 1000 samples across the full set of 8191 codominant loci.

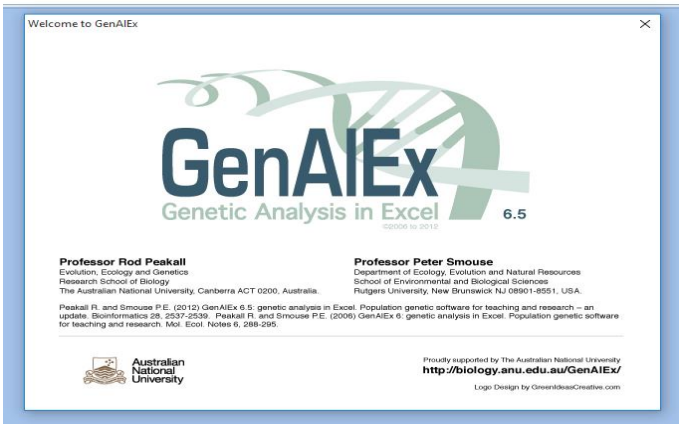

Fig. 6. The GenAlEx Welcom window

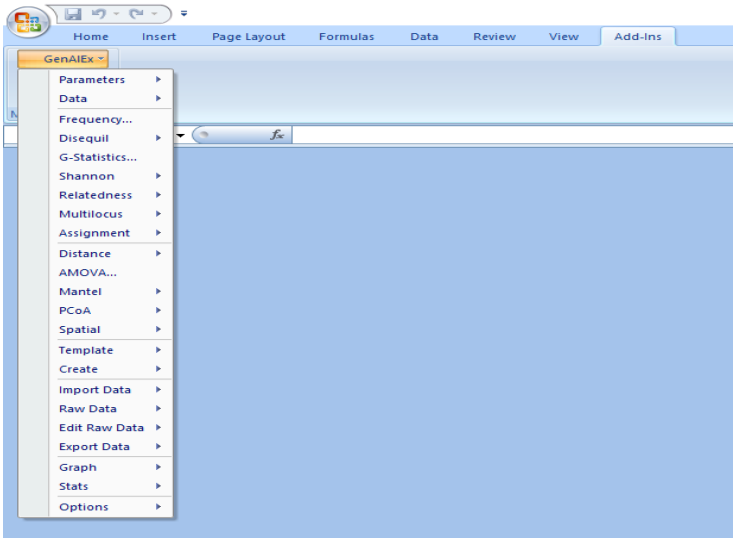

Fig. 7. GenAlEx Menus Options as Add-ins in Excel

GenAlEx requires all data to be coded as numbers and formatted within Excel as numeric data. Be especially careful to avoid using

the text format option, and turn off all auto formatting options. Advanced options are available for processing DNA sequence data to find polymorphisms and haplotypes and convert these to numerical format.

GenAlEx accepts 4 types of numerically-coded data:

1. Codominant genotypic data with 2 columns per locus.
2. Dominant (Binary), Haploid (including Haplotypes), or Sequence data coded numerically with 1 column per locus/base.
3. Codominant and Haploid raw allele frequency data.
4. Geographic data with 2 columns for X and Y coordinates.

ARLEQUIN

Arlequin is the French translation of "Arlecchino", a famous character of the Italian "Commedia dell'Arte". As a character he has many aspects, but he has the ability to switch among them very easily according to its needs and to necessities. This polymorphic ability is symbolized by his colorful costume, from which the Arlequin icon was designed.

The goal of Arlequin is to provide the average user in population genetics with quite a large set of basic methods and statistical tests, in order to extract information on genetic and demographic features of a collection of population samples. The graphical interface is designed to allow users to rapidly select the different analyses they want to perform on their data - Figure 8. Arlequin was written in C++.

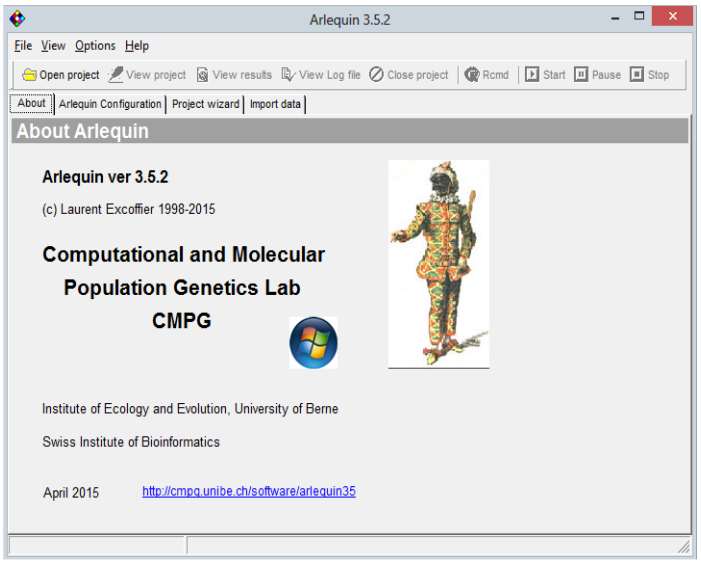

Fig. 8. The Welcome window Interface of Arlequin

We felt important to be able to explore the data, to analyze several times the same data set from different perspectives, with different selected options. The statistical tests implemented in Arlequin have been chosen such as to minimize hidden assumptions and to be as powerful as possible. Thus, they often take the form of either permutation tests or exact tests, with some exceptions.

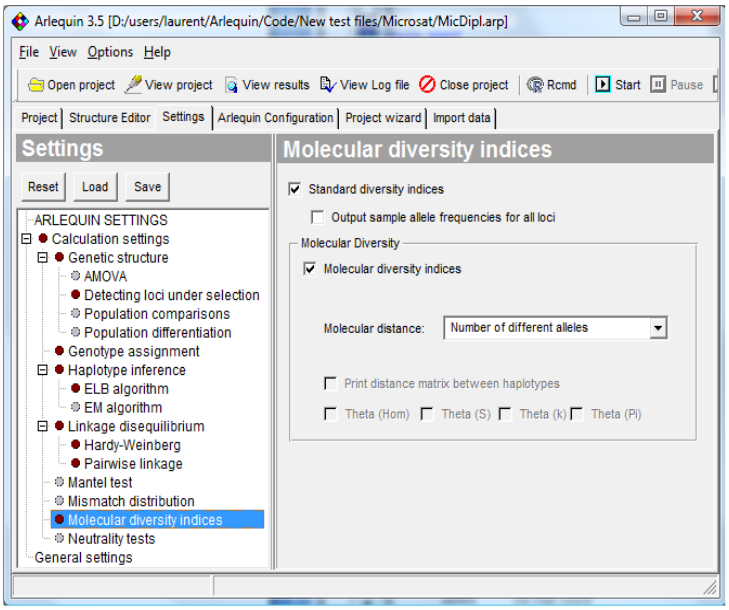

Fig. 9. Arlequin interface showing pick list for selection of modules to run

Arlequin can handle several types of data either in *haplotypic* or *genotypic* form. The basic data types are:

- DNA sequences
- RFLP data
- Microsatellite data
- Standard data
- Allele frequency data

By *haplotypic form* we mean that genetic data can be presented under the form of haplotypes (i.e. a combination of alleles at one or more loci). This haplotypic form can result from the analyses of haploid genomes (mtDNA, Y chromosome, prokaryotes), or from diploid genomes where the gametic phase could be inferred by one way or another.

Because Arlequin has a rich set of features and many options, it means that the user has to spend some time in learning them. However, we hope that the learning curve will not be that steep.

Arlequin is made available free of charge, as long as we have enough local resources to support the development of the program. The analyses Arlequin can perform on the data fall into two main categories: intra-population and inter-population methods. In the first category statistical information is extracted independently from each population, whereas in the second category, samples are compared to each other.

METHODOLOGY

In order to compare the Multiple Sequence Alignment Programs and have a full view of their capabilities, a two-stage comparison process was applied: the "high level" and the "low level" comparison. The former one includes comparisons of the interfaces, the portability, the functionalities and the parameterizations that each of the programs offers, all of them affecting the usability of the program and therefore, its popularity among the users. The latter one compares the "heart" of the programs, the algorithms, that defines the quality and the biological meaning of their results. Therefore, the researcher can choose the program that best fits his/her needs.

Multiple sequence alignments (MSA) are of great importance for biological research. Moreover, the rapid accumulation of DNA sequences during the last years made MSA a necessary tool for research. An MSA can reveal conserved residues that enable the identification of possibly important sites. For example, conserved amino-acid residues are usually involved in protein function or are responsible for protein structural stability. In DNA sequences, conserved regions can represent a regulatory element. Besides of identifying conserved residues a more sophisticated approach is to use information form a MSA by using regions of residues with conserved properties to construct a statistical model such as a Position Specific Scoring Matrix or perhaps a Hidden Markov Model. These models are used to identify conserved regions in newly sequenced genomes, or they are used to construct databases such as PROSITE (Hulo et al., 2004) or PFAM (Bateman et al., 2004)). Sequencing of a whole genome is a difficult task, especially when large eukaryotic genomes are considered. However, one of the main difficulties was raised by the use of MSAs. While the sequencing of a short stretch of DNA is a routine for many molecular labs, it is practically impossible for large sequences. The solution is to cut the sequence in random short stretches and sequence them. The reconstruction of the whole chromosome is done in silico, by aligning the stretches and finding their overlaps.

Another very important application of MSAs is their incorporation in many methods of predicting the structure or function from sequence. These methods are a major contribution of bioinformatics to experimental research. The rate of known sequences is increasing, while other information such as their function is lagging behind.Many methods incorporate information from MSAs to improve their predictions. Such applications are pairwise sequence alignments using structural data (Marti-Renom et al., 2004), recent methods for the prediction of protein secondary structure (Predict Protein, PsiPred, jpred, Baxevanis & Ouellette, 2001) and gene prediction from comparison of sequenced genomes (Brent & Guigó, 2004).

The need of a benchmark study

The number of different MSA methodologies has greatly increased during the last years resulting to approximately 30 programs today. On the contrary only a few of them are used routinely by biologists. The reasons are many, but the main one is the lack of a consistent theoretical framework in sequence analysis (Notredame 2002). As a consequence "programs with badly designed interfaces or poor portability have been discarded by natural selection, leaving their algorithms to be reinvented by later generations" (Notredame 2002). Furthermore, most benchmarks are conducted from researchers that introduce a novel algorithm, and compare only the minority of most used programs. Therefore, a comparison study for the different MSA programs that are offered in the web is necessary, not only for the new scientists that enter the Bioinformatics area, but also for the biologists and bioinformaticians. A more wide and detailed knowledge of all the currently available methods helps scientists to use the proper software that interprets better their biological data and corresponds to their specific biological problem.

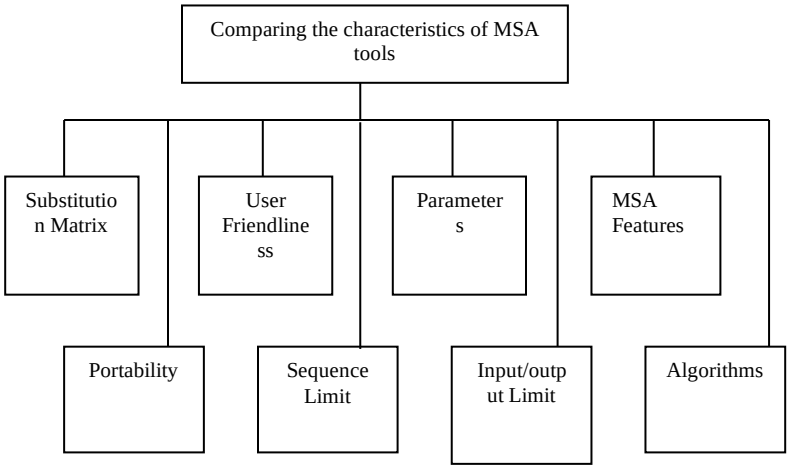

Fig. 10. Comparison structures for the Tools

Module content

Comparison of algorithms

In order to compare the MSA programs in more depth there is a need to study the algorithms. Inyang et al (2019). The performance of the algorithms and the quality of the results is evaluated by comparing the results of a specific MSA program with a "correct" result. But, which multiple alignments are considered as correct? The subjectivity of the reliability of a correct alignment has led to the creation of reference datasets which contain multiple alignments that are generally accepted. And, how two multiple alignments are compared? The quantitative analysis of a comparison between a test multiple alignment and a reference one uses some scores that may are dependent on the alignment itself (e.g. Sum of Pairs, Total Column Score) or independent from it. In the next paragraphs there is a detailed description of the reference datasets, the algorithms and the evaluation scores used in MSA programs.

Algorithms

Two sequences can be aligned either globally or locally, depending on the purpose of study. The global alignment tries to align the sequences at their entire length; therefore it is mostly used for sequences with high similarity in their whole length. Needleman-Wunsch algorithm uses dynamic programming to align globally two sequences allowing the insertion of gaps. Trying to align the sequence at the whole length, may lead to mismatch at local areas. For example, a global alignment of two proteins that share a common domain restricted in their N-terminal region will use information of the whole length of the proteins and perhaps the domains will not be aligned correctly. Smith-Waterman algorithm provides solution to this problem using dynamic programming with an extra condition that result to local alignments. Dynamic programming guarantees the mathematically optimal alignment for a specific score function that is needed to be maximized. In the case of more than two sequences, dynamic programming algorithms can be extended to a multi-dimensional space; therefore, the computational time and memory needed is prohibitive to use for more than three sequences. So, even though dynamic programming algorithms offer the optimal alignment, practically their use is not feasible with the provided current technology and computers. That is why new heuristic approaches have been developed using different strategies.

Exact Algorithms: These algorithms are high quality heuristics that find multiple alignments very close to the optimal. They are based on dynamic programming algorithms, but they exclude from the computation the portion of the multidimensional space that does not contribute to the solution, Inyang et al (2019). In this way the computational time and memory becomes less, offering a less optimal multiple alignment solution. A program called MSA implements this approach and manages to align up to ten closely related sequences in a reasonable computational time (Mount 2001). DCA (Stoye 1997) is a divide and conquer algorithm that uses MSA. DCA algorithm cuts the sequences in subsets of segments that are small enough to be fed to MSA. The critical issue is to cut the sequences at the right points so that the produced alignments remain as close as possible to optimal. DCA manages to align up to 20 – 30 closely related sequences. In the next years with the increase in the computational speed, all the above limitations may not be prohibitive for practical use of exact algorithms in everyday research.

Progressive Algorithms: These algorithms are the most widely used, since they can align multiple sequences in little time and with little memory. Their basic idea is that the final multiple alignment is the result of progressive building upon the alignment of two sequences (or multiple alignments). This means that a progressive assembly of the multiple alignments takes place where the sequences or the alignments are added one by one so that never more than two sequences (or multiple alignments) are simultaneously aligned using dynamic programming. The order of the sequences that are added to the alignment is indicated by a pre-

computed tree, which is computed by aligning pair-wise all against all the sequences. To summarize the whole procedure:

- Align pair-wise all against all the sequences.
- Construction of distance matrix using the pair-wise alignment scores.
- Creation of a distance tree.

Align the two closest sequences. To this alignment add the next closest sequence (or the next closest alignment) and align. Continue with progressive alignments where sequences are added to the multiple alignment according to the order indicated by the tree.

The most widely used progressive programs are ClustalW (Thompson et al., 1994) and T-Coffee (Notredame et al., 2000). Inyang et al (2019)

Iterative Algorithms*:* Iterative alignment methods depend on algorithms able to produce an alignment and to refine it through a series of cycles (iterations) until no more improvements can be made. Iterative methods can be deterministic or stochastic, depending on the strategy used to improve the alignment. The simplest iterative strategies are deterministic. They involve extracting sequences one by one from a multiple alignment and realigning them to the remaining sequences some of these methods can even be a mixture of progressive and iterative strategies. The procedure is terminated when no more improvement can be made (convergence). Stochastic iterative methods include HMM training and simulated annealing or genetic algorithms (Notredame 2002). Widely used iterative programs are Praline (Simossis et al., 2003), PRRP and SAGA (Notredame et al., 1996).

In this study Mega and Arlequin were tested as representative programs for global alignments, while GenAlex was tested as a representative of local alignment programs.

RESULTS

Statistical analysis

For the comparison each methods scores for a specific reference set Friedman test is used (Edgar, 2004, Katoh et al., 2005. Thompson et al., 1999b), which does not make any assumption for the underlying distribution and the only condition is that the samples should have the same size (Lioki-Leivada & Asimakopoulos, 2002). However, the following standardization modules were considered in the research to ascertain the effect of the different tools on a given gene collection. Four different genes were collected for the four tools and the modules in Table 1 tested.

Table 1. Modules for testing with the four selected tools

S/N	Module (see appendix 1)
1	*Standard indices (SI)*
2	*Molecular diversity (MoD)*
3	*Mismatch distribution (MiD)*
4	*Haplotype frequency estimation(Hfe)*
5	*Gametic phase estimation (Gpe)*
6	*Shared haplotypes between populations (sHbp)*
7	*AMOVA*
8	*Pairwise genetic distances (Pgd)*
9	*Exact test of population differentiation (pd)*
10	*Assignment test of genotypes (ToG)*
11	*Detection of loci under selection from F-statistics (Dol)*

Table 2. Photosystem I iron-sulfur center (chloroplast) [Protosiphon botryoides]

GenBank: AYQ95128.1

S/N	Module (see appendix 1)	Mega	GenAlEx	DnaSP	Arlequin
1	*Standard indices (SI)*	23	27.113	31.454	28.56
2	*Molecular diversity (MoD)*	0.865	1.567	0.934	0.774
3	*Mismatch distribution (MiD)*	0.664	0.877	1.231	2.094
4	*Haplotype frequency estimation(Hfe)*	56	44	67	55
5	*Gametic phase estimation (Gpe)*	24	25	37	22
6	*Shared haplotypes between populations (sHbp)*	2.3	3.1	2.4	2.8
7	*AMOVA*	45	42	46	39
8	*Pairwise genetic distances (Pgd)*	1.945	2.687	2.567	1.734
9	*Exact test of population differentiation (pd)*	35	46	63	53
10	*Assignment test of genotypes (ToG)*	23	25	36	27
11	*Detection of loci under selection from F-statistics (Dol)*	15.52	17.37	14.73	16.29

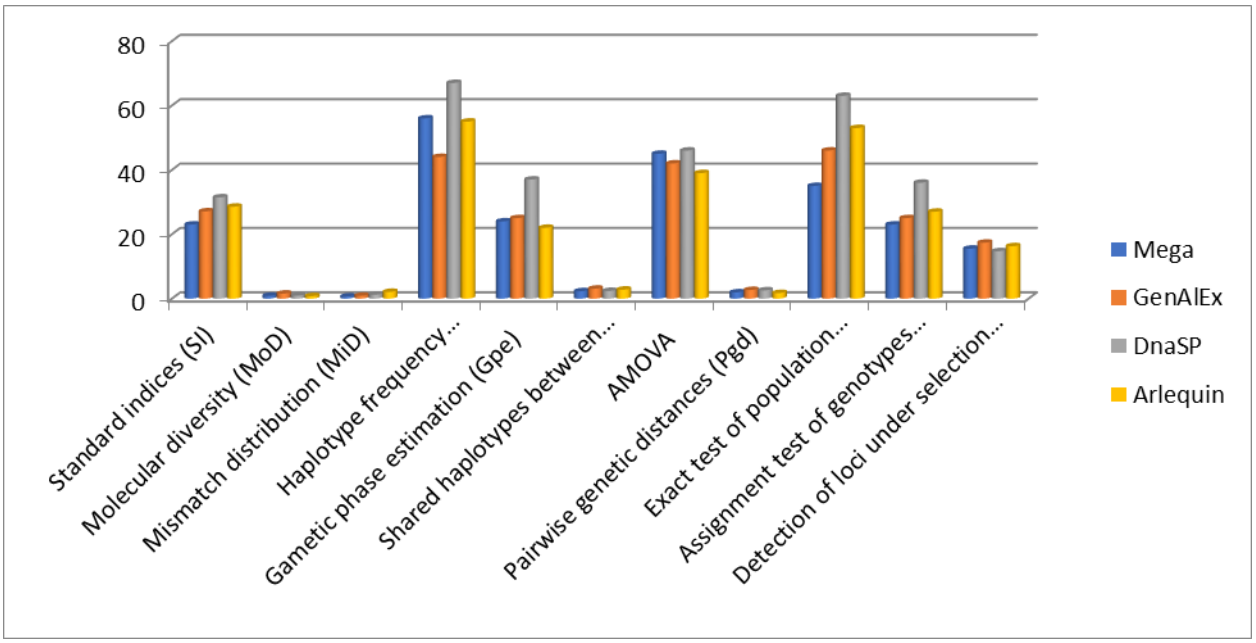

Fig. 11. Graph of Table 2

Table 3. ATP-binding subunit of protochlorophyllide reductase (chloroplast) [Protosiphon botryoides]

GenBank: AYQ95126.1

S/N	Module (see appendix 1)	Mega	GenAlEx	DnaSP	Arlequin
1	*Standard indices (SI)*	14.765	23.113	14.762	25.77
2	*Molecular diversity (MoD)*	0.657	0.422	0.723	0.521
3	*Mismatch distribution (MiD)*	0.117	0.661	0.455	3.243
4	*Haplotype frequency estimation(Hfe)*	22	52	45	71
5	*Gametic phase estimation (Gpe)*	17	10	15	18
6	*Shared haplotypes between populations (sHbp)*	0.24	0.41	0.33	0.22
7	*AMOVA*	34.2	36.2	35.1	30.3
8	*Pairwise genetic distances (Pgd)*	0.945	0.687	0.567	0.734
9	*Exact test of population differentiation (pd)*	24	25.6	24.8	33
10	*Assignment test of genotypes (ToG)*	0.002	0.261	0.005	0.003
11	*Detection of loci under selection from F-statistics (Dol)*	16.222	15.843	16.401	14.994

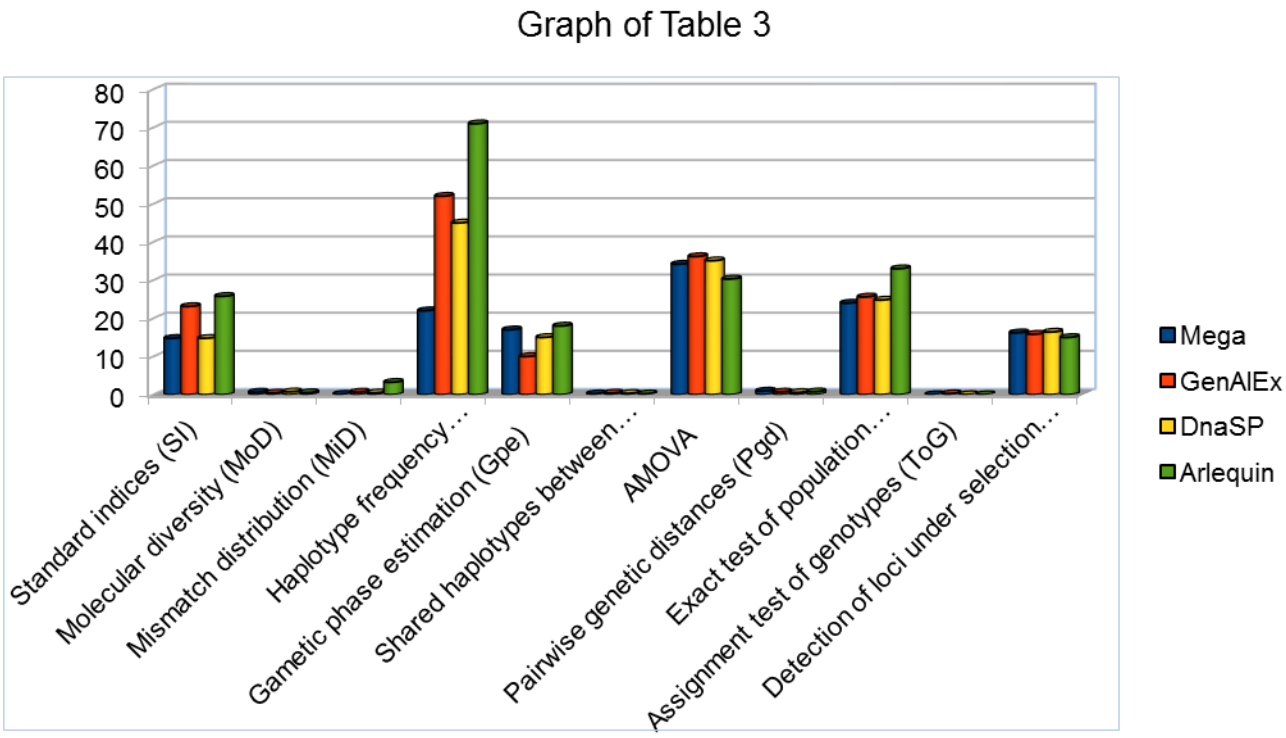

Fig. 12. Graph of Table 3

CONCLUSION

Comparison and alignment of a series of protein and DNA sequences were among the first and are now established as the most powerful and frequently used bioinformatics methods. A variety of computational algorithms and programs has been created for this purpose. Inyang et al (2019). Decision about which tools to use is one of the important problems for bioinformaticians, especially for the majority of biologists who are non-specialist users. Therefore, a comparisons study for the different multiple sequence alignment tools (MSA) is necessary for the biologists and bioinformaticians to use the proper software that interprets correctly their biological data. This study addresses this critical issue in relation to MSA algorithms by systematically comparing and evaluating the functionality, usability and the algorithms of the above four famous multiple sequence alignment tools.

REFERENCES

Alkan,C. et al. (2009) Personalized copy number and segmental duplication maps using next-generation sequencing. Nat. Genet., **41**:1061–1067.

Bateman, A., Coin, L., Durbin, R., Finn, R.D., Hollich, V., Griffiths-Jones, S., Khanna, A., Marshall, M., Moxon, S., Sonnhammer, E.L.L, Studholme, D.J., Yeats, C., Eddy, S. 2004. The Pfam protein families database. Nucleic Acids Research. 32 (Suppplement 1):138-141

Baxevanis, A., Ouellette, B.F.F. 2001. Bioinformatics : A practical guide to the analysis of genes and proteins (2nd ed). John Wiley & Sons, Inc.

Edgar, R. 2004. MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Research, 32(5):1792-1797.

Horner,D.S. et al. (2010) Bioinformatics approaches for genomics and post genomics applications of next-generation sequencing. Brief. Bioinformatics, **11**:181–197.

Hulo, N., Sigrist, C.J.A., Le Saux, V. Langendijk-Genevaux, P.S, Bordoli, L., Gattiker,
A., De Castro, E., Bucher, P, Bairoch, A. 2004. Recent improvements to the PROSITE database. Nucleic Acids Research, 32 (Supplement 1):134-137

Katoh, K., Kuma, K., Toh, H., Miyata, T. (2005). MAFFT version 5: improvement in accuracy of multiple sequence alignment. Nucleic Acids Research, 33(2):511-518.

Katoh, K., Misawa, K., Kuma, K,. Miyata, T. (2002). MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Research, 30(14):3059-3066.

Lioki-Leivada, I, Asimakopoulos, D.N. (2002). Introduction to applied statistics. V.1. University of Athens.

Marti-Renom, M.A., Madhusudhan, M.S., Sali, A. (2004). Alignment of protein sequences by their profiles. Protein Science 13:1071-1087

Medvedev,P. et al. (2009) Computational methods for discovering structural variation with next-generation sequencing. Nat. Methods, **6**: S13–S20.

Miller,J.R. et al. (2010) Assembly algorithms for next-generation sequencing data. Genomics, **95**, 315–327.

Mount, D.W., (2001). Bioinformatics, Sequence and Genome Analysis. Cold Spring Harbor Laboratory Press.

Notredame, C., (2002). Recent progress in multiple sequence alignment: a survey. Pharmacogenomics 3(1):131-144.

Notredame, C., Higgins, D.G, Heringa, J. (2000). T-Coffee: a novel method for fast and accurate multiple sequence alignment. J. Mol. Biol. 302:205-217.

Notredame, C., Higgins, D.G. 1996. SAGA: sequence alignment by genetic algorithm. Nucleic Acids Research 24(8): 1515–1524.

Novocraft (2010) http://www.novocraft.com/. (last accessed date October 28, 2010).

Qin,J. et al. (2010) A human gut microbial gene catalogue established by metagenomic sequencing. Nature, **464**, 59–65.

Inyang G. A, Ogban F. U. Udensi U.O, (2019) Efficacy of the Algorithm(S) In Analytical Software Packages on DNA Sequence Data Analysis.

Simossis, V.A., Heringa, J. (2003). The PRALINE online server: optimising progressive multiple alignment on the web. Computational Biology and Chemistry 27:511–519.

Stoye, J. (1997). Divide-and-Conquer Multiple Sequence Alignment. Dissertation Thesis. University of Bielefeld, Forschungsbericht der Technischen Fakultat, Abteilung Informationstechnik

Thompson, J.D., Plewniak, F., Poch, O. (1999)b. A comprehensive comparison of multiple sequence alignment programs. Nucleic Acids Research, 27(13):2682-2690.

Thompson, J.D., Higgins, D.G., Gibson, T.J. (1994). CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Research 22(22):4673-4680.

Appendix 1.

Intra-population methods:	***Short description:***
Standard indices (SI)	*Some diversity measures like the number of polymorphic sites, gene diversity.*
Molecular diversity (MoD)	*Calculates several diversity indices like nucleotide diversity, different estimators of the population parameter θ.*
Mismatch distribution (MiD)	*The distribution of the number of pairwise differences between haplotypes, from which parameters of a demographic (NEW) or spatial population expansion can be estimated*
Haplotype frequency estimation(Hfe)	*Estimates the frequency of haplotypes present in the population by maximum likelihood methods.*
Gametic phase estimation (Gpe)	*Estimates the most like gametic phase of multi-locus genotypes using a pseudo-Bayesian approach (ELB algorithm).*
Inter-population methods:	***Short description:***
Shared haplotypes between populations (sHbp)	*Comparison of population samples for their haplotypic content. All the results are then summarized in a table.*
AMOVA	*Different hierarchical **A**nalyses of **Mo**lecular **Va**riance to evaluate the amount of population genetic structure.*
Pairwise genetic distances (Pgd)	*FST based genetic distances for short divergence time.*
Exact test of population differentiation (pd)	*Test of non-random distribution of haplotypes into population samples under the hypothesis of panmixia.*

Assignment test of genotypes (ToG)	*Assignment of individual genotypes to particular populations according to estimated allele frequencies.*
Detection of loci under selection from F-statistics (Dol)	*Detection of loci under selection by the examination of the joint distribution of FST and heterozygosity under a hierarchical island model.*