MQAT: An Efficient Quality Assessment Tool for Large Multiple Sequence Alignments

Muhammad Tariq Pervez[1,3], Masroor Ellahi Babar[2], Asif Nadeem[3], Naeem Aslam[4,3], Ali Raza Awan[3], Muhammad Aslam[4], Tanveer Hussain[3], Salman Qadri[6], Sarfraz Ahmad[1] and Muhammad Shoaib[4]

[1]Department of Computer Science, Virtual University of Pakistan
[2]Department of Live Stock Production, University of Veterinary and Animal Sciences, Lahore, Pakistan
[3]Institute of Biochemistry and Biotechnology, University of Veterinary and Animal Sciences, Lahore, Pakistan
[4]Department of Computer Science and Engineering, University of Engineering and Technology, Lahore, Pakistan.
[5]Department of Computer Science, NFC Institute of Engineering & Technological Training, Multan, Pakistan
[6]Department of Computer Science, Islamia University of Bahawalpur
tariq_cp@hotmail.com

Abstract: Knowledge of the most accurate MSA method in the initial stage of a biological research work may help in choosing the right MSA method and a correct downstream analysis. The most important challenge of the current era is to handle large alignments efficiently. Currently, no tool is available that compares several large MSAs (having several thousand sequences) simultaneously and efficiently based on a reference alignment. In this article, we present MQAT; a multithreaded java based software tool that can compare several MSAs simultaneously and efficiently. It has implemented divide and conquer technique. MQAT is many times more efficient than the available tools for comparing MSAs. Results show that MQAT can compute sum of pairs score and column score of an alignment consisting of more than 11,000 sequences just in 11 seconds and is more than 95% efficient as compared to other similar tools. All activities in MQAT can be saved in form of a project in an XML file that can be reopened at some later time. MQAT presents results in tabular form as well as in graphical form using bar, pie and line charts.
[Muhammad Tariq Pervez, Masroor Ellahi Babar, Asif Nadeem, Naeem Aslam, Ali Raza Awan, Muhammad Aslam, Tanveer Hussain, Salman Qadri, Sarfraz Ahmad and Muhammad Shoaib. **MQAT: An Efficient Quality Assessment Tool for Large Multiple Sequence Alignments.** *Life Sci J* 2013;10(9s):9-16] (ISSN:1097-8135). http://www.lifesciencesite.com. 2

Keywords: MSA, Comparison, SPS, CS

1. Introduction

Multiple sequence alignments (MSA) have significant role in the downstream analysis which includes identifying (i) conserved patterns through evolution[1,2,3] (ii) functionally important residues, (iii) annotation of novel genomes (iv) prediction of protein secondary and tertiary structure and the nsSNPs (non synonymous Single Nucleotide Polymorphisms) that have a basic role for altering a protein function[4,5,6,7,8]. Several areas of bioinformatics and evolutionary biology are based on correct MSA[9], which is thus, one of the most active and highly scrutinized areas of research in bioinformatics[10,11]. The more correct MSA, the more accurate results of downstream analysis will be.

High throughput sequencing approaches are generating megabase long sequences at an enormous rate [12,13]. Genome sequence alignment tools such as MUMmer[14], GS-Aligner[15], Avid[16] and LAGAN[17], and MSA methods like Clustal W/X[18], T-COFFEE[19], MAFFT[20], Kalign[21], MultiPip-Maker[22], MULTIZ[23], MLAGAN[17], MAVID[24], and MUSCLE[25] can generate alignments consisting of thousands of sequences of several kilobase long. Firstly, efficient computation of these large alignments is a big challenge and it is a heavily scrutinized domain of bioinformatics. Secondly, these tools are heuristics based which do not provide optimal solution and have some deficiencies in one or the other way [26]. Therefore, knowledge of the most accurate MSA method in the initial stage of a biological research work may help in choosing the right MSA method for the right situation [27] and to perform correct downstream analysis. Measuring quality of a MSA method involves calculation of two most commonly used scores i.e. Sum of Pairs Score (SPS) and Column Score (CS) [28]. An alignment having greater SPS and CS is said to be more accurate and vise versa.

A number of bioinformatic tools for comparing MSAs based on reference alignment are available. Examples of such tools are SinicView [29], AltAVist [11], SuiteMSA [10] and a program written in c language by developers of BAliBASE [30]. SinicView can compare multiple nucleotide alignments under a fixed window. SinicView provides both graphical as well as text view for comparison purpose; however, it is not efficient for large alignments. AltAVist is a web based tool for comparing two alignments. Conserved as well as reliably aligned regions are color coded for visualizing local agreement between two alignments.

Due to a web based tool, it imposes restrictions on the size of alignments. SuiteMSA is a good graphical tool for comparing multiple alignments. In addition to SPS and CS, it provides numerous other statistics as a part of comparison between a test and reference alignment. It also provides graphical interface for indel-seq-gen, a molecular evolution simulation tool [31]. However, due to heavy graphics involved, SuiteMSA is not good for large alignments. Program provided by BAliBASE developers is a command based tool and can compute scores of only one alignment at a time. All these tools except SinicView allow user to provide one test and one reference alignment at a time. None of these tools provide facility to save the work done in form of a project.

In this article, we introduce MQAT, a multithreaded bioinformatic tool written in java programming language. MQAT provides a graphical interface to select several test alignments against a single reference alignment. It implements an algorithm based on divide and conquer technique and is many times more efficient than the available tools for comparing MSAs. Results show that MQAT can compute sum of pairs and column score of an alignment consisting of more than 11,000 sequences just in 11 seconds and is more than 95% efficient as compared to other similar tools. MQAT divides an MSA into a number of sub MSAs and assign the job of calculating scores to the threads. Each thread reports its performance to the main thread that calculates final scores. SPS and CS of all test alignments are displayed in tabular form. Results can be sorted by test files, SPS or CS. MQAT allows user to view graphical comparison summary of the selected test alignments in form of bar, pie and line charts. MQAT also provides facility to edit title of the charts, labels of x-axis and y-axis and labels of parts of the charts. It also provides facility to save whole work done in XML format. Saving whole work performed, enables a user to open and continue the work from saved state. MQAT also provides features of printing data in PDF, HTML and on a paper. It also has feature to export data to MS Excel.

2. Martials and Methods

MQAT: Multithreaded Algorithm

The core feature of MQAT is 'Divide and Conquer' approach which is implemented by a very powerful feature of multithreading of java programming language. MQAT algorithm calculates number of threads based on the number of sequences in an MSA. Minimum two threads are generated for every alignment and this number is incremented for every other 500 hundred sequences. MQAT algorithm used the following equation to compute the number of threads required for an alignment.

$$N_t = \frac{N_s}{500} + 2 \qquad (1)$$

Where N_t is number of threads required for an MSA. N_s represent number of sequences in an alignment and '2' denotes the minimum number of threads required for every alignment.

MQAT algorithm divides an MSA horizontally into sub MSAs based on the number of threads generated by equation 1. Equation 2 and 3 calculate number of sequences for the sub MSAs. Number of sequences for the first sub MSA is calculated by equation 2 and equation 3 is executed repeatedly (until N_t reaches to zero) to compute number of sequences for subsequent sub MSAs.

$$N_{si} = N_s / N_t \qquad (2)$$

$$\begin{bmatrix} N_t = N_t - 1 \\ N_{sj} = \dfrac{N_s - N_{si}}{N_t} \\ N_{si} = N_{si} + N_{sj} \end{bmatrix} \qquad (3)$$

In equation 2 N_{si} denotes number of sequences of the first sub MSA that is to be generated whereas in equation 3 it is the total number of sequences to extracted from N_s. N_{sj} represent number of sequences of the current sub MSA to be generated.

After calculating required number of threads and generating sub MSAs, MQAT creates threads and assign them the task of calculating SPS and CS for the sub MSAs. Main thread gets scores of the sub MSAs form the respective threads and computes the final scores. The whole sequence of the algorithm is shown by figure 1.

MQAT: The Tool

MQAT is written in java programming language using java development kit 7u15, and netbeans 7.3. In this section, we present usage of major interfaces of MQAT.

Start New Project Interface

In MQAT, all activities have to be part of a project. A project is a collection of test files along with their SPS and CS and other analysis activities such as bar, pie or line charts of the selected test files. A new project is created by clicking the 'New Project' button in 'File Menu' of the main window of MQAT (Figure 2). Interface of starting new project provides facility to select multiple test files and compute their SPS and CS against a single reference file. A user can either remove all files by selecting 'Reset List' button or any selected file from the list by clicking 'Remove File' button. By pressing 'Start' button, MQAT starts computing SPS and CS of all provided files and 'Cancel' button may be used to cancel the activity.

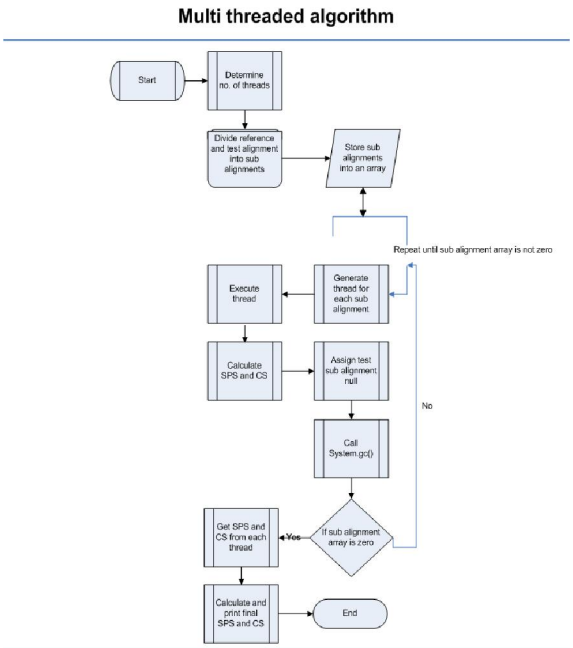

Figure 1. Graphical representation of multithreaded algorithm. In the start, algorithm takes both reference and test alignment. After calculating number of threads required for each test alignment, it divides the alignment into sub alignments. The algorithm, then, executes threads for computing SPS and CS for each sub alignment. At the end, main thread computes final score and prints them on screen.

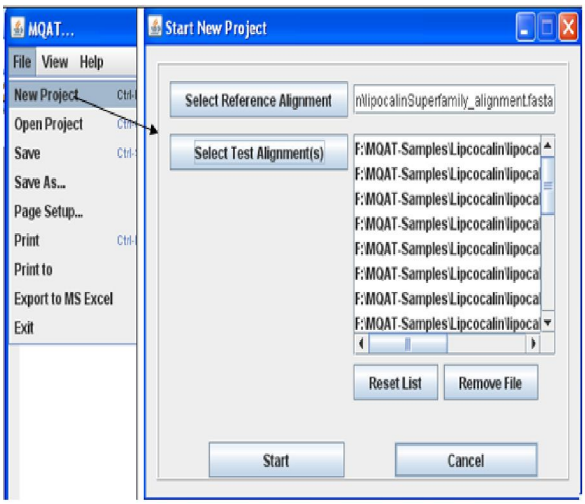

Figure 2. The window to start a new project (right hand side) is opened by clicking 'New Project' button in file menu of the main window of MQAT (left hand side). A user can select multiple test files, remove selected or all files. 'Start' button starts a new project and 'Cancel' button cancels the project.

Scores Interface

When user presses 'Start button' on 'Start New Project' window, MQAT begins calculating SPS and CS of all loaded test files and displays results in tabular form in a new sub window (Figure 3) inside the main window. Results are displayed under three labels/columns i.e. 1) 'Sort by Test Files', 2) 'Sort by SPS' and 'Sort by Column Score'. Column labeled as 'Sort by Test Files' shows all processed test files and other two columns shows SPS and CS respectively. The prefix 'Sort by' with label of each column means that user can sort results in ascending or descending order by clicking these buttons with respect to test files, SPS or column score. Options to select the desired alignments are also provided for further analysis in form of bar, pie or line charts.

Figure 2 displays SPS and CS of lipocalin superfamily proteins generated from various MSA methods. These are a group of small globular proteins and in addition to other functions; they are mostly associated in allergic reactions. They also share a common antiparallel beta-barrel conformation consisting of eight beta-strands. Apart from this, lipocalin proteins have a small highly-conserved motif near the first beta- strand[32,33]. We obtained both the manually-adjusted MSA from Sánchez et al. [32]. Strope et al. [12] and Catherine et al.[10] used the same proteins to illustrate their tools.

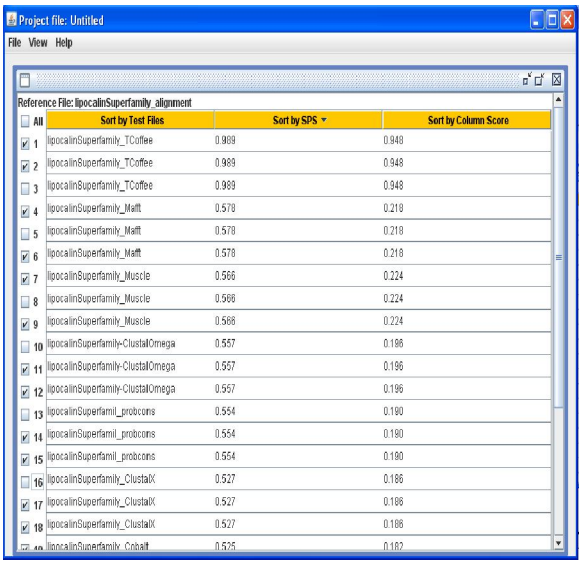

Figure 3. A sub window inside the main window displaying SPS and CS. The button labeled as 'Sort by Test Files' provides feature of sorting results in ascending or descending order with respect to test files. The buttons of 'Sort by SPS' and 'Sort by Colum Score' sort results with respect to SPS and CS respectively. Extreme left pane provides options to select desired alignments for further analysis. The figure shows results in descending order with respect to SPS.

Graphical Analysis Tools

MQAT provides three major graphical tools for analysis of accuracy of alignments. These are bar, pie and line charts both for SPS and CS. Bar and pie charts are calculated based on the total values of SPS and CS while line charts are built based on per column SP and CS.

Bar Chart

Bar charts are one of the three graphical tools for analysis. Figure 4 shows bar chart analysis of accuracy of lipocalin superfamily proteins whose SPS and CS is shown in figure 2. Window displaying bar chart also provides several editing features such as a user may edit title of the chart, labels of x-axis and y-axis and labels of individual bars of the chart. 'Chart Update Form' is displayed when a user clicks with mouse on any of the bars of the chart. After providing new label in the text field and pressing the 'Ok' button, old label of the selected bar is replaced by the new one.

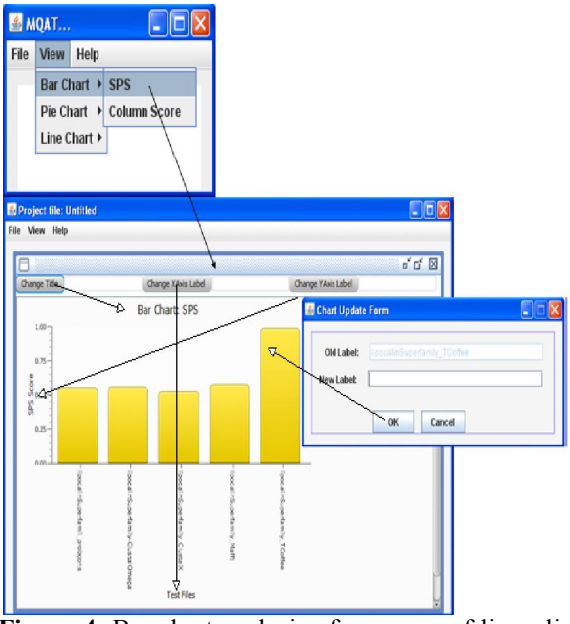

Figure 4. Bar chart analysis of accuracy of lipocalin superfamily protein alignment.

Pie Chart

Analysis of accuracy of alignments can also be viewed via pie chart. Figure 5 shows pie chart analysis of accuracy of lipocalin superfamily proteins shown in figure 5. This window also provides editing features like bar chart window. A user can edit title of chart and labels of individual parts of pie chart. "Chart Update Form" is displayed when user clicks inside of a part of pie chart. Remaining procedure is similar to "Chart Update Form" displayed in bar chart window.

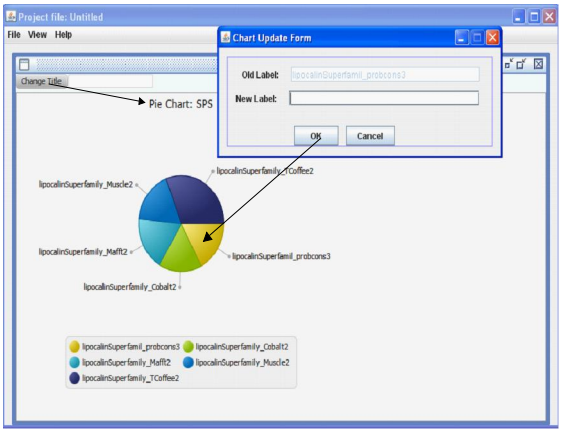

Figure 5. Pie chart analysis of the test alignments. This window also provides options to edit pie chart title and labels of parts of pie chart.

Line Chart

Line chart (Figure 6) displays graphical analysis of SPS and CS of each column of the alignments. Like bar and pie chart, window of line chart provides various editing options. Title of line chart, label of x-axis, y-axis and line itself can be edited and changed. Upper pane of line chart window provides options to edit title of chart and labels of x-axis and y-axis. Bottom pane is for editing label of a line.

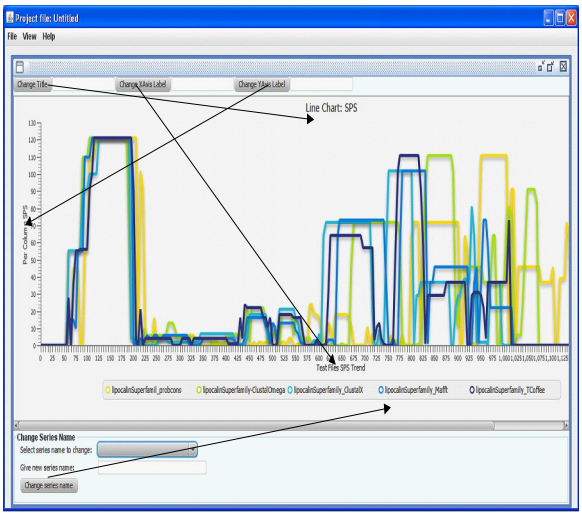

Figure 6. Per column SPS view of the selected alignments in figure 2. Upper part of this window displays options to edit title of the chart, labels of x-axis and y-axis while the lower part titled as "Change Series Name" provides an interface to edit labels of the lines (inside the legend).

3. Results and Discussion

One of the techniques of examining accuracy of MSA methods is to compare an alignment constructed by the MSA method (called as

test alignment) with a reference alignment [34-35]. Two popular scores i.e. SPS and CS are calculated as a part of this comparative study. Currently, available tools calculate scores only of one test alignment at a time which is a time consuming activity and it becomes a very tedious task when you want to compare several test alignments. MQAT gives an option to provide test files as many as you want and displays SPS and CS in tabular form of all the provided alignments. Multithreaded algorithm has enabled MQAT to handle large alignments efficiently. Accuracy of MQAT is comparable to SuiteMSA and BAliBASE c program and, in case of large alignments especially, efficiency is very high as compared to SuiteMSA and BAliBASE c program. A reference alignment can be obtained by three approaches. Firstly, it can be get from a bench mark MSA database, secondly by adjusting an MSA by hand based on our own experience and knowledge and thirdly, it can be constructed by using a simulator such as ROSE [35], iSG [31], MySSP [36] Seq-Gen [37] and SIMPROT [38].

In this section, we discuss performance of MQAT with three angles i.e. 1) Accuracy 2) Efficiency with respect to other tools and 3) Efficiency for larger alignments.

Accuracy of MQAT

In this section, we show that efficiency of MQAT does not effect its accuracy. We take lipocalin superfamily protein alignments computed by different MSA methods for the accuracy comparison of the three tools. Figure 7 shows that values of SPS and CS computed by MQAT are very similar to the values calculated by SuiteMSA and BAliBASE 'c' program. SuiteMSA is good for small alignments, BAliBASE 'c' program is good for small as well as medium alignments but MQAT is good for small, medium as well as large alignments, therefore, in this section we have chosen a small protein data set so that we may present comparison of all the three tools.

Efficiency: BAliBASE 'c' program vs. MQAT

In this section, we show efficiency comparison of MQAT and BAliBASE 'c' program. Due to limitation of BAliBASE 'c' program, we have shown comparison of alignments having up to one thousand sequences. Results show that MQAT spends almost 5 seconds where as BAliBASE 'c' program consumes 159 seconds to calculate scores of alignments shown in figure 8. It means that MQAT is about 97% more efficient as compared to BAliBASE 'c' program. Data used in this comparison is taken from BAliBASE.

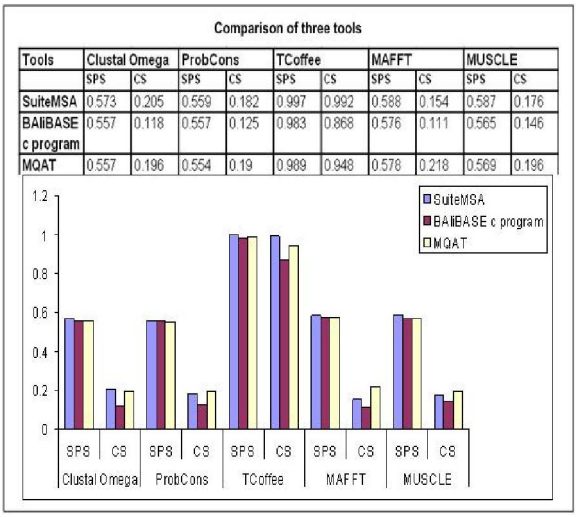

Figure 7. Comparison of SuiteMSA, BAliBASE 'c' program and MQAT with respect to accuracy. The figure shows that accuracy of MQAT is very close to SuiteMSA and BAliBASE 'c' program.

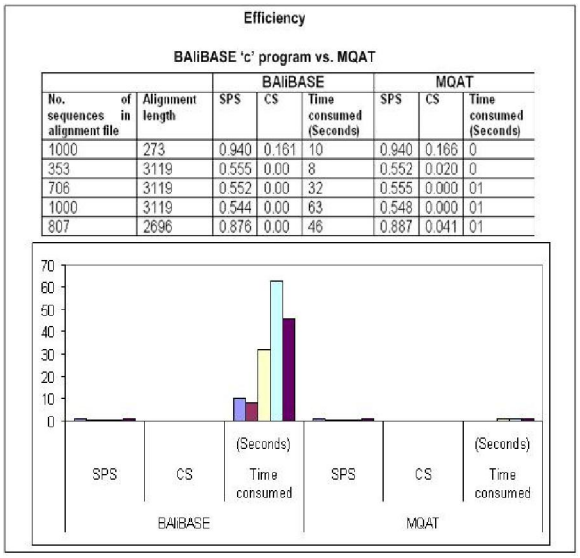

Figure 8. Comparison of efficiency between BAliBASE 'c' program and MQAT. Significant difference of time consumed by BAliBASE 'c' program and MQAT can be seen.

Efficiency of MQAT for larger alignments

MQAT is also very efficient for alignments having more than 1000 sequences. This section presents efficiency statistics of MQAT for alignments of varying number of sequences and length. MQAT computes scores of alignment having 1614 sequences just in 2 and 11298 sequences just in 12 seconds. Alignments for the purpose of analysis shown in figure 10 were constructed by using Clustal Omega from the sequence file named as BBA0039.tfa in folder titled as RV100 enclosed in a zipped file

named as 'msa_reference.tar.gz' in BAliBASE and then replicating it to generate an alignment having 1614 sequences and then replicating 1614 sequences to generate 3228 and so on. Replication was done in order to save time and avoid from constructing so big alignments from an MSA method.

Time consumed by MQAT per alignment

No. of sequences	SPS	CS	Time consumed (Seconds)
1614	0.887	0.041	2
3228	0.887	0.041	4
4842	0.882	0.025	6
6456	0.883	0.028	8
8070	0.884	0.031	10
9684	0.882	0.024	11
11298	0.877	0.00	12

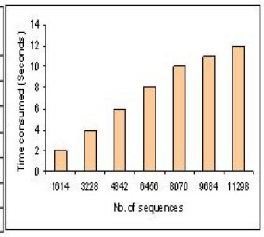

Figure 10. Efficiency analysis of MQAT for alignments having more than one thousand sequences. The figure shows that MQAT consumes just 53 seconds to compute scores of all alignments comprising of 45192 sequences in total.

4. Conclusions

MQAT allows users to calculate SPS and CS of large alignments very quickly. Other tools either don't accept alignment consisting of more than one thousand sequences or they are very slow. MQAT has been tested for an alignment consisting of more than eleven thousand sequences but it is expected that MQAT should work for even larger alignments. Results show that MQAT is more than 95% efficient as compared to other similar tools. MQAT can calculate scores of multiple MSAs simultaneously. It shows results in text format in tabular form where user is allowed to sort results in ascending or descending order with respect to test files, SPS or CS. MQAT also allows user to perform graphical analysis of the selected alignments. Graphical analysis can be made using bar, pie and line charts. Bar and pie charts provides analysis for total SPS and CS whereas line charts analyze SPS and CS of each column of the alignment. Various parameters of these graphical tools can be edited by the user. MQAT also provides facility to save the work done in form of a project in XML format as well as to open it at any time in future. MQAT also permits users to print results in HTML, PDF format or on a paper.

Availability and requirements

• Project name: MQAT
• Project home page: http://www.ivistmsa.com/
• Operating system(s): MS Windows
• Programming language: java 1.7
• License: none
• Any restrictions to use by non-academics: none

Additional Material

- Alignment files constructed by various MSA methods of 23 lipocalin protein sequences are included.
- A reference file of 23 lipocalin protein sequences is included.
- Test and reference alignments of sequences taken from BAliBASE for the purpose of efficiency comparison between MQAT and BAliBASE 'c' program are included.
- To show efficiency of MQAT, test and reference alignments of sequences taken from BAliBASE is include.

Authors' contributions

Muhammad Tariq Pervez conceived the idea, developed the tool and drafted the paper. Masroor Ellahi Babar, developed the system organization, and drafted some parts of the paper. Asif Nadeem and Ali Raza supervised the system implementation and also contributed in organization of the paper. Muhammad Shoaib, Muhammad Aslam Naeem Aslam and Tanveer Hussain contributed in implementation of the tool.

Acknowledgements

We thank Higher Education Commission of Pakistan for her generous support for completing this project successfully. We also thank Dr. Shahzad Ahmad Faizi, lecturer, Department of Mathematics and Mr. Imran Ali, lecturer, Department of English, Virtual University of Pakistan for valuable discussions and help in drafting the paper.

Corresponding Authro

Muhammad Tariq Pervez
Department of Computer Science
Virtual University of Pakistan
E-mail: tariq-cp@hotmail.com

References

1. Kim J, Ma J. PSAR: measuring multiple sequence alignment reliability by probabilistic sampling. Nucl. Acids Res. 39 (15):6359-6368.doi:10.1093/nar/gkr334. .2011.
2. Siepel A, Bejerano G, Pedersen JS, Hinrichs AS, Hou M, Rosenbloom K,Clawson H, Spieth J, Hillier LW, Richards S, Weinstock GM, Wilson RK, Gibbs RA, Kent WJ, Miller W, Haussler D: Evolutionarily conserved elements in vertebrate, insect, worm, and yeast genomes. Genome Research 2005, 15(8):1034-1050.
3. Roskin KM, Diekhans M, Haussler D: Scoring Two-Species Local Alignments to Try to Statistically Separate Neutrally Evolving from Selected DNA Segments. Proceedings of the

seventh annual international conference on Computational molecular biology ACM Press; 2003, 257-266[http://doi.acm. org/10.1145/640075.640109].
4. Waterhouse AM., Procter, JB., Martin DMA, Clamp M, Barton GJ. Jalview version 2: A Multiple Sequence Alignment and Analysis Workbench, Bioinformatics doi: 10.1093/bioinformatics/btp033. 2009.
5. Thompson JD, Linard B, Lecompte O, Poch O. A Comprehensive Benchmark Study of Multiple Sequence Alignment Methods: Current Challenges and Future Perspectives. PLoS ONE 6(3): e18093. doi:10.1371/journal.pone.0018093. 2011.
6. Sullivan OO, Zehnder M, Higgins D, Bucher P, Grosdidier A, Notredame C. APDB: a novel measure for benchmarking sequence alignment methods without reference. Bioinformatics. 2003. 19:1. i1215-i1221.
7. Katoh K, Kuma K, Toh H. MAFFT v. 5: improvement in accuracy of multiple sequence alignment. Nucleic. Acids Res. 2005. 33, 511–518.
8. Reinert, K. et al. An iterative methods for faster sum-of-pairs multiple sequence alignment. Bioinformatics. 2000. 16, 808–814.
9. Rausch, T. et al. Genome sequencing of pediatric medulloblastoma links catastrophic DNA rearrangements with TP53 mutations. Cell. 2012. 148, 59–71.
10. Catherine LA, Cory LS, Etsuko NM. SuiteMSA: Visual Tools for Multiple Sequence Alignment Comparison and Molecular Sequence Simulation. BMC Bioinformatics. 2011. 12:184.
11. Morgenstern B, Goel S, Sczyrba A, Dress, A. AltAVisT: comparing alternative multiple sequence alignments. Bioinformatics. 2003. 19, 425–426.
12. Roskin et al.: Meta-Alignment with Crumble and Prune: Partitioning very large alignment problems for performance and parallelization. BMC Bioinformatics. 2011 12:144.
13. Liolos K, Tavernarakis N, Hugenholtz P, Kyrpides N: The Genomes On Line Database (GOLD) v.2: a monitor of genome projects worldwide. Nucleic Acids Research. 2006. 34:D332-334.
14. Delcher AL, Kasif S, Fleischmann RD, Peterson J, White O, Salzberg SL: Alignment of whole genomes. Nucleic Acids Research. 1999. 27(11):2369-2376
15. Shih AC, Li WH: GS-Aligner: a novel tool for aligning genomic sequences using bit-level operations. Mol Biol Evol. 2003. 20(8):1299-1309.
16. Bray N, Dubchak I, Pachter L: AVID: A global alignment program. Genome Research. 2003. 13(1):97-102.
17. Brudno M, Do CB, Cooper GM, Kim MF, Davydov E, Green ED, Sidow A, Batzoglou S: LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. Genome Research. 2003. 13(4):721-731.
18. Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ, HigginsDG. Clustal W and Clustal X version 2.0. Bioinformatics. 2007. 23, 2947-2948
19. Notredame C, Higgins DG, Heringa J: T-Coffee: A novel method for fast and accurate multiple sequence alignment. J Mol Biol. 2000. 302(1):205-217.
20. Katoh K, Misawa K, Kuma K, Miyata T: MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Res. 2002. 30(14):3059-3066.
21. Lassmann T, Frings OS, Sonnhammer, EL. Kalign 2: high-performance multiple alignment of protein and nucleotide sequences allowing external features. Nucleic Acids Res. 2009. 37: 858–865.
22. Sánchez D, Ganfornina MD, Gutiérrez G, Marín A: Exon-intron structure and evolution of the lipocalin gene family. Mol Biol Evol. 2003. 20:775-783.
23. Blanchette M, Kent WJ, Riemer C, Elnitski L, Smit AF, Roskin KM, Baertsch R, Rosenbloom K, Clawson H, Green ED, Haussler D, Miller W: Aligning multiple genomic sequences with the threaded blockset aligner. Genome Res. 2004. 14(4):708-715.
24. Bray N, Pachter L: MAVID: constrained ancestral alignment of multiple sequences. Genome Res. 2004. 14(4):693-699.
25. Edgar RC: MUSCLE: a multiple sequence alignment method with reduced time and space complexity. BMC Bioinformatics. 2004. 5(1):113.
26. Notredame C. Recent evolutions of multiple sequence alignment algorithms. PLoS Comput. Biol., 2007. 3, e123.
27. Lassmann T, Sonnhammer E: Automatic assessment of alignment quality. Nucleic Acids Research. 2005. 33: 7120-7128.
28. Yasuo T, Hisanori K, Taishin K and Kiyoshi Asai. A fast structural multiple alignment method for long RNA sequences. BMC Bioinformatics. 2008. 9:33 doi:10.1186/1471-2105-9-33.

29. Shih AC, Lee DT, Lin L, Peng CL, Chen SH, Wu YW, Wong CY, Chou MY, Shiao TC, Hsieh MF. SinicView: a visualization environment for comparisons of multiple nucleotide sequence alignment tools. BMC Bioinformatics. 2006. 7:103.
30. Thompson J, Plewniak F, Poch O: BAliBASE: a benchmark alignment database for the evaluation of multiple alignment programs.Bioinformatics. 1999. 15:87-8.
31. Strope CL, Abel K, Scott SD, Moriyama EN. Biological sequence simulation for testing complex evolutionary hypotheses: indel-Seq-Gen version 2.0. Mol Biol Evol. 2009. 26:2581-2593.
32. Flower DR, North ACT, Attwood TK: Structure and sequence relationships in the lipocalins and related proteins. Protein Sci. 1993. 2:753-761.
33. Sánchez D, Ganfornina MD, Gutiérrez G, Marín A: Exon-intron structure and evolution of the lipocalin gene family. Mol Biol Evol. 2003. 20:775-783.
34. Changhoon K, Byungkook L. Accuracy of structure-based sequence alignment of automatic methods. BMC Bioinformatics. 2007. 8:355 doi:10.1186/1471-2105-8-355
35. Stoye J, Evers D, Meyer F. Rose: generating sequence families. Bioinformatics. 1998. 14:157–163.
36. Rosenberg MS: MySSP: non-stationary evolutionary sequence simulation, including indels. Evol Bioinform Online. 2005. 1:81-83.
37. Rambaut A, Grassly NC. Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. Comput Appl Biosci. 1997.13:235–238.
38. Pang A, Smith AD, Nuin PAS, Tillier ERM. SIMPROT: using an empirically determined indel distribution in simulations of protein evolution BMC Bioinformatics. 2005. 6: 236.

6/28/2013