A Survey of Multiple Sequence Alignment Techniques

Xiao-Dan Wang(✉), Jin-Xing Liu, Yong Xu, and Jian Zhang

Bio-Computing Research Center, Shenzhen Graduate School,
Harbin Institute of Technology, Heilongjiang, China
{wangxiaodan0608, zpower007}@163.com,
sdcavell@126.com, yongxu@ymail.com

Abstract. Multiple sequence alignment (MSA) is a basic step in many bioinformatics analyses, and also a NP-hard problem. In order to improve the speed, accuracy and cater to the requirement of large-scale sequences alignment, a wide variety of MSA methods and softwares have been subsequently developed. In this article, we will systematically review the wildly used methods and introduce their practical results on the benchmark Balibase 3.0 references. We come to the conclusion that computational complexity still is the bottleneck of MSA. We also consider future development of MSA methods with respect to applying of more different technologies and the prospect of parallelization of MSA.

Keywords: Multiple sequence alignment · MSA techniques

1 Introduction

With the further rapid development of new sequencing technology, the biological applications become more and more widely, including exposition of relationship between nucleosome positioning and DNA methylation [1], prediction of missense mutation or protein functionality [2, 3], the assembly of new genomes [4], crop breeding [5], and so on. For most of these applications, multiple sequence alignments are fundamental.

For N sequences of length L, the exact way of computing an optimal alignment has a computational complexity of $O(N^L)$, which is excessive even for small number of sequences. Unfortunately, all sequencing technologies in production, such as Illumina, Helicos, SOLiD and Roche/454, can produce thousands or millions of sequences concurrently [6, 7]. In order to overcome this difficulty, many heuristic methods, including progressive methods [8] and iterative refinement methods [9] are developed.

This article aims to systematically review the recent advance of MSA methods. It is organized as follows. We first introduce the basic theory of heuristic methods and review the development of wildly used techniques, including Clustal, T-Coffee, MAFFT, MUSCLE and Kalign in Sect. 2, and then examine their programs on the benchmark Balibase 3.0 references [10], Oxbench [11] and Homestrad in Sect. 3. Finally, we discuss the future development of multiple sequence alignment in Sect. 4.

D.-S. Huang et al. (Eds.): ICIC 2015, Part I, LNCS 9225, pp. 529–538, 2015.
DOI: 10.1007/978-3-319-22180-9_52

2 Overview

2.1 Theory

Progressive Method. The progressive method is the first practical MSA construction strategy, and still composes the key of a majority of MSA programs by now. A progressive method usually is made up of four steps as follows [12]:

Step 1: Calculate a distance matrix for N input sequences. The element of this matrix is the distance of every pair of the input sequences, and there are many ways to messure distance, for example, angle cosine and Euclidean distance. In a exact way, $\binom{N}{2}$ pair-wise alignments are needed to count the numbers of matches, mismatches, and indels, which are then converted to the distance measures. This procedure is costly when N is large, as its time complexity is $O(N^2L^2)$;

Step 2: Construct a guide tree according to the distance matrix calculated in Step1 by a clustering analysis method. The most widely used method is UPGMA (Unweighted Pair-Group Method with Arithmetic means) [13] which takes computation time of $O(N^2)$ to construct the guide tree;

Step 3: In the guide tree, an external node represents each input sequence, while an internal node represents an MSA;

Step 4: Repeat Step1 and Step2 for the generated pair-wise alignments after construction of the initial MSA.

Iterative Refinement. The progressive method is implemented using a "greedy algorithm" by what mistakes made at the initial alignment stages cannot be corrected later [14]. To overcome this defect, an effective approach relies on post process known as iterative refinement, which also consists of four steps as follow [12]:

Step 1: Construct an initial MSA;

Step 2: Divide the MSA constructed in Step1 into two groups, then get rid of the columns made up of nulls from each of the two groups;

Step 3: Realign the two groups produced in Step2 by a pair-wise sequence-to-group or group-to-group alignment method;

Step 4: Repeat Step2 and Step3 until no gain in the alignment score or the iterative times exceeding a predefined number.

Scoring Function. A good scoring function is necessary to guarantee this procedure work accurately. The most widely used function is sum-of-pairs (SP) score [15] and weighted sum-of-pairs score (WSP) [16] with affine gaps.

For a sequence set A which is made up of N sequences of length L, we define WSP as follow:

$$WSP(A) = \sum_{1 \le i < j \le N} w_{i,j} H(a_i, a_j) = \sum_{1 \le l \le L} \sum_{1 \le i < j \le N} w_{i,j} [S(a_{i,l}, a_{j,l}) - v \cdot G(i,j,l)], \tag{1}$$

where $H(a_i, a_j)$ is the alignment score of a pair of sequences in A, $w_{i,j}$ is the weight corresponding to the pair sequences $[a_i, a_j]$ ($w_{i,j} = 1$ is an unweighted case), $S(a_{i,l}, a_{j,l})$ is the match score of the pair sequences $[a_i, a_j]$ at position l, $G(i,j,l)$ is a Boolean variable which is defined as follows, if a gap opens between a_i and a_j at position l, $G(i,j,l) = 1$, else $G(i,j,l) = 0$, and v is the penalty of gap.

2.2 Alignment Technique

Clustal. In 1988, the first Clustal program was written by Des Higgins [17], and a dynamic programming algorithm [18] and the progressive alignment strategy developed by Feng and Doolittle [8] were combined in this program. It used a word-based alignment algorithm [19] to calculate the distance matrix and UPGMA method was used to construct the guide tree. In 1992, ClustalV [20] implemented profile alignments to generate guide trees from the multiple alignment using the Neighbour-Joining (NJ) method [21]. In 1994, ClustalW [22] improved the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. In 1997, ClustalX [23] provided a visual interface, so that the multiple alignment can be displayed on the screen and all parameters were optional, which was a significant convenience to the user's of evaluation. The latest member of Clustal series program is Clustal Omega [14], which can align virtually any number of protein sequences quickly and delivers accurate alignments. For constructing a guide tree, Clustal Omega uses a modified version of mBed [24] which has complexity of $O(N \log N)$ and the guide tree is just as accurate as those from conventional methods. In Clustal Omega, the alignments are then computed using the very accurate HHalign package [25], which aligns two profile hidden Markov models [17].

T-Coffee. The first T-Coffee (Tree-based Consistency Objective Function for alignment Evaluation) [26] version can be track back to 2000. It implemented progressive alignment with a consistency-based objective function [27] and tried to maximize the score between the final multiple alignment and a library of pair-wise aligned residue scores which is derived from a mixture of local and global pair-wise alignments. M-Coffee [28] is an extension of T-Coffee and uses consistency to estimate a consensus alignment, and a meta-method for assembling multiple sequence alignments (MSA) by combining the output of several individual methods into one single MSA. TCS (Transitive Consistency Score) [29] is a new extended version of the T-Coffee scoring scheme for overcoming the problem that homology and evolutionary modeling are sensitive to the underlying MSA accuracy, and it also can improve phylogenetic tree reconstruction.

MAFFT. MAFFT [30] was a method for rapid multiple protein sequence alignment based on FFT (Fast Fourier Transform), first released in 2002. Homologous region

were rapidly identified by the FFT. FFT converted an amino acid sequence to a sequence whose composition were volume and polarity values of each amino acid residue. The original MAFFT included two different heuristics, the progressive methods were FFT-NS-1 and FFT-NS-2 and the iterate refinement method was FFT-NS-i. In 2005, MAFFT version 5 [31] was released with improvement of accuracy by offering new iterative refinement options, H-INS-i, F-INS-i and G-INS-i. And MAFFT version 5 incorporated pair-wise alignment information into objective function. In 2007, MAFFT version 6 [32] improved accuracy of multiple ncRNA alignment with two techniques: the PartTree algorithm and the Four-way consistency objective function. In 2010, for speeding up program, two natural parallelization strategies (best-first and simple hill-climbing) were implemented for the iterative refinement stage based on MAFFT version 6, and a simple hill-climbing approach was selected as the default [33]. In 2012, two methods had been implemented as the '–add' and '–addfragments' options in the MAFFT package [34] for adding unaligned sequences into an existing multiple sequence alignment.

The newest version is MAFFT version 7 [35], it has options for adding unaligned sequences into an existing alignment, and beyond this, it has several new features, including adjustment of direction in nucleotide alignment, constrained alignment and parallel processing.

MUSCLE. MUSCLE (MUltiple Sequence Comparison by Log-Expectation) [36] is a multiple sequence alignment method of protein sequences. MUSCLE uses two distance measures for each pair of sequences: a kmer distance (for an unaligned pair) and the Kimura distance (for an aligned pair). Guide tree is constructed using UPGMA. MUSCLE uses a profile function called log-expectation (LE) score. And MUSCLE includes three stages as follow:

Stage 1: Draft progressive. This stage includes four steps (similarity measure, distance estimate, tree construction, progressive alignment) and produces a rapid multiple alignment, while de-emphasizing accuracy.

Stage 2: Improved progressive. This stage also includes four steps (similarity measure, tree construction, tree comparison, progressive alignment). In the stage1, the main source of error is the k-mer distance measure, which leads to a suboptimal tree. MUSCLE therefore re-estimates the tree using the Kimura distance, which is more accurate but requires an alignment.

Stage 3: Refinement. This stage is made up of four steps (choice of bipartition, profile extraction, re-alignment, accept/reject). The third stage performs iterative refinement using a approximate tree-dependent restricted partitioning [21].

Kalign. Kalign [31] was a MSA algorithm, which proposed in 2005. It also implemented progressive alignment. And unlike other progressive methods, Kalign employed Wu-Manber approximate string-matching algorithm [37] which made Kalign more accurate in aspect of distance estimation. In 2007, Emmanuelle Becher etc. proposed a tool called HMM-Kalign [38] for generating sub-optimal alignments. As the name implies, HMM-Kalign was based on original Kalign by implementing Hidden Markove Model. The newest inproved edition of Kalign was Kalign-LCS [39].

It applied the longest common subsequence (LLCS) in similarity measure step, and obtained a balance between accuracy and speed.

3 Practical Result

We examine ClustalW, Clustal Omega, T-Coffee, MAFFT:Auto, MAFFT:FFT-NS-1, MAFFT:G-INS-i, MUSCLE and Kalign on the benchmark Balibase 3.0 references, OXbench and Homestrad, respectively.

We evaluate the alignment results with BaliScore, including SP-score (Sum of Pairs score) which is the percentage of homologies in the reference alignment recovered in the estimated alignment and TC-score (Total column score) is the percentage of columns that are recovered entirely correctly in the estimated alignment (Tables 1, 2 and 3).

Table 1. Summary of the techniques described in the review

Name		Method	Guide tree	Sequence	Server
ClustalW [22]		Progressive	NJ	Protein DNA	http://www.clustal.org/clustal2/ http://www.ebi.ac.uk/Tools/msa/clustalw2/
Clustal Omega [14]		Progressive	mBed, PartTree	Protein DNA RNA	http://www.clustal.org/omega/ http://www.ebi.ac.uk/Tools/msa/clustalo/
T-Coffee [26]		Progressive	–	Protein DNA RNA	http://www.tcoffee.org/ http://www.ebi.ac.uk/Tools/msa/tcoffee/
MAFFT [35]	***FFT-NS-1***	Progressive	PartTree	Protein DNA RNA	http://mafft.cbrc.jp/alignment/server/ http://www.ebi.ac.uk/Tools/msa/mafft/
	FFT-NS-2				
	G-INS-1				
	FFT-NS-i	Iterative refinement			
	E-INS-i				
	L-INS-i				
	G-INS-i				
	Q-INS-i				
MUSCLE [36]	***Step1*** ***Step2***	Progressive	UPGMA	Protein	http://www.drive5.com/muscle/ http://www.ebi.ac.uk/Tools/msa/muscle/
	Step3	Iterative refinement			
Kalign		Progressive	Wu-Manber	Protein DNA RNA	http://msa.sbc.su.se/cgi-bin/msa.cgi http://www.ebi.ac.uk/Tools/msa/kalign/

Table 2. The SP-score of various individual methods on the benchmark Balibase 3.0 references

SP-score	ClustalW	Clustal Omega	T-Coffee	MAFFT: Auto	MAFFT: FFT-NS-1	MAFFT: G-INS-i	MUSCLE	Kalign
BaliBase Set: 11	98.7 %	99.3 %	100 %	82.9 %	62.7 %	88.4 %	90.4 %	91.2 %
BaliBase Set: 12	97.3 %	100 %	100 %	93.1 %	87.8 %	94.4 %	90.3 %	90.5 %
BaliBase Set: 20	43.5 %	93.8 %	95.6 %	42.9 %	36.9 %	48.0 %	47.6 %	70.1 %
BaliBase Set: 30	61.4 %	70.7 %	94.4 %	63.7 %	63.6 %	66.4 %	60.8 %	65.0 %
BaliBase Set: 40	93.4 %	93.3 %	97.4 %	90.7 %	88.4 %	90.6 %	90.3 %	88.7 %
BaliBase Set: 50	69.1 %	71.4 %	84.7 %	63.5 %	58.3 %	66.5 %	58.7 %	61.7 %
Average of BaliBase	77.2 %	88.1 %	95.4 %	66.3 %	66.3 %	75.7 %	73.0 %	77.9 %
OXbench set: full	0	100 %	100 %	2.0 %	1.7 %	2.3 %	2.4 %	1.0 %
OXbench set: master	7.7 %	73.2 %	100 %	6.9 %	6.1 %	9.0 %	7.0 %	7.1 %
OXbench set: extended	8.1 %	11.7 %	96.2 %	7.4 %	7.3 %	8.3 %	8.8 %	7.8 %
Average of OXbench	5.3 %	61.6 %	98.7 %	5.4 %	5.0 %	6.5 %	6.1 %	5.3 %
Homestrad	96.9 %	95.1 %	99.1 %	82.7 %	75.1 %	83.9 %	77.9 %	77.3 %

Table 3. The TC-score of various individual methods on the benchmark Balibase 3.0 references

TC-score	ClustalW	Clustal Omega	T-Coffee	MAFFT: Auto	MAFFT: FFT-NS-1	MAFFT: G-NS-i	MUSCLE	Kalign
BaliBase Set: 11	97.4 %	98.7 %	100 %	76.3 %	42.1 %	80.3 %	85.5 %	84.2 %
BaliBase Set: 12	94.2 %	100 %	100 %	86.5 %	78.8 %	88.5 %	80.8 %	80.8 %
BaliBase Set: 20	0	85.9 %	88.5 %	0	0	0	0	0
BaliBase Set: 30	22.7 %	33.1 %	79.1 %	22.7 %	22.1 %	23.3 %	23.9 %	23.0 %
BaliBase Set: 40	61.5 %	58.1 %	80.8 %	50.6 %	38.5 %	48.3 %	45.3 %	40.4 %
BaliBase Set: 50	24.4 %	27.7 %	0	16.5 %	13.5 %	19.0 %	12.7 %	12.2 %
Average of BaliBase	50.3 %	67.3 %	74.7 %	42.1 %	32.5 %	43.2 %	41.4 %	40.1 %
OXbench set: full	0	100 %	100 %	0	0	0	0	0
OXbench set: master	0	50.0 %	100 %	0	0	0	0	0
OXbench set: extended	0	0	83.7 %	0	0	0	0	0
Average of OXbench	0	50.0 %	94.6 %	0	0	0	0	0
Homestrad	91.8	87.0 %	95.5 %	53.0 %	31.4 %	57.2 %	44.2 %	31.5

From the results of SP-score and TC-score, we can see that all programs we examined are not sensitive to divergence of sequence. All programs suffer by the impact of a highly divergent “orphan” sequence, residue difference between groups, N/C-terminal extensions, and internal insertions to varying degrees, respectively. And on the whole, Clustal Omega and T-Coffee perform well, especially the results corresponding to T-Coffee are the best.

4 Conclusion and Future Development

In the past years, MSA achieved great development, and obtained good effect which applied in many biological applications. But there still is plenty room to improve multiple sequence alignment, especially in the respect of robustness and accuracy. In order to solve these problems, in one hand, we should continue to develop recent efficient MSA techniques, such as T-Coffee, in other hand we should transform the way of thinking and apply more techniques which are not just heuristic methods, even not just biological informatics technology to improve MSA.

Happily, many researchers devote themselves to develop MSA method. Sabari Pramanik and S.K. Setua [40] define a new form of chromosome representation, and deploy it on steady state Genetic Algorithm, then get better results. Siavash Mirarab, Nam Nguyen, and Tandy Warnow propose an algorithm called PASTA [41] to realize estimation of large-scale multiple sequence alignment. And there is a interesting method called Phylo [42], which is a human-based computing framework applying "crowd sourcing" techniques to solve the Multiple Sequence Alignment (MSA) problem. The key idea of Phylo is to convert the MSA problem into a casual game that can be played by ordinary web users with a minimal prior knowledge of the biological context. Cactus [43] caters to the phenomenon that much attention has been given to the problem of creating reliable multiple sequence alignments in a model incorporating substitutions, insertions, and deletions while far less attention has been paid to the problem of optimizing alignments in the presence of more general rearrangement and copy number variation.

Another trend of development is parallelization of MSA. Because of that MSA is a NP-hard problem and the huge amount of data, the programs of MSA are costly in the respect of time. Hence, it's necessary to implement parallel solutions in MSA. Jucele F. A. et al. [44] present two parallel solutions using the BSP/CGM model, with MPI and CUDA implementations. And the results of this method show that the use of parallel processing allows the manipulation of more and larger sequences. Evandro A. Marucci et al. [45] propose a parallel algorithm for multiple sequence similarities calculation based on the k-mer counting method, and obtain a very good scalability and a nearly linear speedup.

Acknowledgement. This work was supported by Shenzhen Municipal Science and Technology Innovation Council (Grant No. CXZZ20140904154910774, Grant No.JCYJ20140417172417174, Grant No. JCYJ20140904154645958, Grant No. JCYJ20130329151843309) and China Postdoctoral Science Foundation funded project (Grant No. 2014M560264).

References

1. Chodavarapu, R.K., Feng, S., Bernatavichute, Y.V., Chen, P.-Y., Stroud, H., Yu, Y., et al.: Relationship between nucleosome positioning and DNA methylation. Nature **466**, 388–392 (2010)

2. Hicks, S., Wheeler, D.A., Plon, S.E., Kimmel, M.: Prediction of missense mutation functionality depends on both the algorithm and sequence alignment employed. Hum. Mutat. **32**, 661–668 (2011)
3. Wang, P., Hu, L., Liu, G., Jiang, N., Chen, X., Xu, J., et al.: Prediction of antimicrobial peptides based on sequence alignment and feature selection methods. PLoS one **6**, e18476 (2011)
4. Brenchley, R., Spannagl, M., Pfeifer, M., Barker, G.L., D'Amore, R., Allen, A.M., et al.: Analysis of the bread wheat genome using whole-genome shotgun sequencing. Nature **491**, 705–710 (2012)
5. Varshney, R.K., Terauchi, R., McCouch, S.R.: Harvesting the promising fruits of genomics: applying genome sequencing technologies to crop breeding. PLoS Biol. **12**, e1001883 (2014)
6. Li, H., Homer, N.: A survey of sequence alignment algorithms for next-generation sequencing. Briefings Bioinform. **11**, 473–483 (2010)
7. Zhou, X., Ren, L., Meng, Q., Li, Y., Yu, Y., Yu, J.: The Next-generation sequencing technology and application. Protein Cell **1**, 520–536 (2010)
8. Feng, D.-F., Doolittle, R.F.: Progressive sequence alignment as a prerequisitetto correct phylogenetic trees. J. Mol. Evol. **25**, 351–360 (1987)
9. Hogeweg, P., Hesper, B.: The alignment of sets of sequences and the construction of phyletic trees: an integrated method. J. Mol. Evol. **20**, 175–186 (1984)
10. Thompson, J.D., Koehl, P., Ripp, R., Poch, O.: BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. Proteins Struct. Funct. Bioinf. **61**, 127–136 (2005)
11. Raghava, G., Searle, S.M., Audley, P.C., Barber, J.D., Barton, G.J.: OXBench: a benchmark for evaluation of protein multiple sequence alignment accuracy. BMC Bioinf. **4**, 47 (2003)
12. Gotoh, O.: Heuristic Alignment Methods. Multiple Seq. Alignment Meth. **1079**, 29–43 (2014)
13. Kersters, K., De Ley, J., Sneath, P., Sackin, M.: Numerical taxonomic analysis of agrobacterium. J. Gen. Microbiol. **78**, 227–239 (1973)
14. Sievers, F., Wilm, A., Dineen, D., Gibson, T.J., Karplus, K., Li, W., et al.: Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. Mol. Syst. Biol. **7**, 539 (2011)
15. Altschul, S.F.: Gap costs for multiple sequence alignment. J. Theor. Biol. **138**, 297–309 (1989)
16. Altschul, S.F., Carroll, R.J., DJ, L.: Weights for Data Related by a Tree. J. Mol. Biol. **207**, 647–653 (1989)
17. Eddy, S.R.: Profile hidden markov models. Bioinformatics **14**, 755–763 (1998)
18. Myers, E.W., Miller, W.: Optimal alignments in linear space. Comput. Appl. Biosci. CABIOS. **4**, 11–17 (1988)
19. Wilbur, W.J., Lipman, D.J.: Rapid similarity searches of nucleic acid and protein data banks. Proc. Natl. Acad. Sci. **80**, 726–730 (1983)
20. Higgins, D.G.: CLUSTAL V: multiple alignment of DNA and protein sequences. Comput. Anal. Seq. Data **25**, 307–318 (1994)
21. Saitou, N., Nei, M.: The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol. Biol. Evol. **4**, 406–425 (1987)
22. Thompson, J.D., Higgins, D.G., Gibson, T.J.: CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res. **22**, 4673–4680 (1994)
23. Thompson, J.D., Gibson, T.J., Plewniak, F., Jeanmougin, F., Higgins, D.G.: The CLUSTAL_X windows Interface: Flexible Strategies for Multiple Sequence Alignment Aided by Quality Analysis Tools. Nucleic Acids Res. **25**, 4876–4882 (1997)

24. Blackshields, G.S.F., Shi, W., Wilm, A., Higgins, D.G.: Sequence embedding for fast construction of guide trees for multiple sequence alignment. Algorithms Mol Biol. **5**, 21 (2010)
25. Söding, J.: Protein homology detection by HMM–HMM comparison. Bioinformatics **21**, 951–960 (2005)
26. Notredame, C., Higgins, D.G., Heringa, J.: T-Coffee: a novel method for fast and accurate multiple sequence alignment. J. Mol. Biol. **302**, 205–217 (2000)
27. JD, K.: The maximum weight trace problem in multiple sequence alignment. In: Apostolico, A., Crochemore, M., Galil, Z., Manber, U. (eds.) CPM 1993. LNCS, vol. 684, pp. 106–119. Springer, Heidelberg (1993)
28. Wallace, I.M., O'Sullivan, O., Higgins, D.G., Notredame, C.: M-Coffee: combining multiple sequence alignment methods with t-coffee. Nucleic Acids Res. **34**, 1692–1699 (2006)
29. Chang, J.-M., Di Tommaso, P., Notredame, C.: TCS: A New Multiple Sequence Alignment Reliability Measure to Estimate Alignment Accuracy and Improve Phylogenetic Tree Reconstruction. Molecular Biology and Evolution. msu117(2014)
30. Katoh, K., Misawa, K., K.-I, K., Miyata, T.: MAFFT: a novel method for rapid multiple sequence alignment based on fast fourier transform. Nucleic Acids Res. **30**, 3059–3066 (2002)
31. Katoh, K., Kuma, K.-i, Toh, H., Miyata, T.: MAFFT Version 5: improvement in accuracy of multiple sequence alignment. Nucleic Acids Res. **33**, 511–518 (2005)
32. Katoh, K., Toh, H.: Improved accuracy of multiple ncRNA alignment by incorporating structural information into a MAFFT-based framework. BMC Bioinform. **9**, 212 (2008)
33. Katoh, K., Toh, H.: Parallelization of the MAFFT multiple sequence alignment program. Bioinform. **2**, 1899–1900 (2010)
34. Katoh, K., Frith, M.C.: Adding unaligned sequences into an existing alignment using MAFFT and LAST. Bioinform. **28**, 3144–3146 (2012)
35. Katoh, K., Standley, D.M.: MAFFT multiple sequence alignment software Version 7: improvements in performance and usability. Mol. Biol. Evol. **30**, 772–780 (2013)
36. Edgar, R.C.: MUSCLE: multiple aequence alignment with high accuracy and high throughput. Nucleic Acids Res. **32**, 1792–1797 (2004)
37. Wu, S., Manber, U.: Fast text searching: allowing errors. Commun. ACM **35**, 83–91 (1992)
38. Becker, E., Cotillard, A., Meyer, V., Madaoui, H., Guérois, R.: HMM-Kalign: a tool for generating sub-optimal HMM alignments. Bioinform. **23**, 3095–3097 (2007)
39. Deorowicz, S., Debudaj-Grabysz, A., Gudyś, A.: Kalign-LCS — a more accurate and faster variant of kalign2 algorithm for the multiple sequence alignment problem. In: Gruca, A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3. AISC, vol. 242, pp. 499–506. Springer, Heidelberg (2014)
40. Pramanik, S., Setua, S.: A steady state genetic algorithm for multiple sequence alignment. In: International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 1095–1099. IEEE (2014)
41. Mirarab, S., Nguyen, N., Warnow, T.: PASTA: ultra-large multiple sequence alignment. In: Sharan, R. (ed.) RECOMB 2014. LNCS, vol. 8394, pp. 177–191. Springer, Heidelberg (2014)
42. Kawrykow, A., Roumanis, G., Kam, A., Kwak, D., Leung, C., Wu, C., et al.: Phylo: a citizen science approach for improving multiple sequence alignment. PLoS one **7**, e31362 (2012)
43. Paten, B., Earl, D., Nguyen, N., Diekhans, M., Zerbino, D., Haussler, D.: Cactus: algorithms for genome multiple sequence alignment. Genome Res. **21**, 1512–1528 (2011)

44. Vasconcellos, J.F., Nishibe, C., Almeida, N.F., Cáceres, E.N.: Efficient parallel implementations of multiple sequence alignment using BSP/CGM model. In: Proceedings of Programming Models and Applications on Multicores and Manycores, 103. ACM (2014)
45. Marucci, E.A., Zafalon, G.F., Momente, J.C., Neves, L.A., Valêncio, C.R., Pinto, A.R. et al.: An Efficient Parallel Algorithm for Multiple Aequence Aimilarities Calculation Using a Low Complexity Method. BioMed research international (2014)