Bioinformatics, 34(14), 2018, 2490–2492
doi: 10.1093/bioinformatics/bty121
Advance Access Publication Date: 1 March 2018
Applications Note

Sequence analysis

Parallelization of MAFFT for large-scale multiple sequence alignments

Tsukasa Nakamura[1,2], Kazunori D. Yamada[2,3], Kentaro Tomii[1,2,4,5] and Kazutaka Katoh[2,6,*]

[1]Department of Computational Biology and Medical Sciences, Graduate School of Frontier Sciences, University of Tokyo, Chiba 277-8562, Japan, [2]Artificial Intelligence Research Center (AIRC), National Institute of Advanced Industrial Science and Technology (AIST), Tokyo 135-0064, Japan, [3]Graduate School of Information Sciences, Tohoku University, Sendai 980-8579, Japan, [4]Biotechnology Research Institute for Drug Discovery (BRD), AIST, Tokyo 135-0064, Japan, [5]AIST-Tokyo Tech Real World Big-Data Computation Open Innovation Laboratory (RWBC-OIL), Tokyo 152-8550, Japan and [6]Research Institute for Microbial Diseases, Osaka University, Suita 565-0871, Japan

*To whom correspondence should be addressed.
Associate Editor: John Hancock

Received on October 19, 2017; revised on February 7, 2018; editorial decision on February 24, 2018; accepted on February 28, 2018

Abstract

Summary: We report an update for the MAFFT multiple sequence alignment program to enable parallel calculation of large numbers of sequences. The G-INS-1 option of MAFFT was recently reported to have higher accuracy than other methods for large data, but this method has been impractical for most large-scale analyses, due to the requirement of large computational resources. We introduce a scalable variant, G-large-INS-1, which has equivalent accuracy to G-INS-1 and is applicable to 50 000 or more sequences.

Availability and implementation: This feature is available in MAFFT versions 7.355 or later at https://mafft.cbrc.jp/alignment/software/mpi.html.

Contact: katoh@ifrec.osaka-u.ac.jp

Supplementary information: Supplementary data are available at *Bioinformatics* online.

A large number of biological sequences from widely divergent organisms are becoming available. Accordingly, the need for multiple alignments of large numbers of sequences is increasing for various kinds of sequence analysis. The G-INS-1 option of MAFFT was recently reported to have higher accuracy than other methods for large multiple sequence alignments (MSAs) in independent benchmarks (Le *et al.*, 2017; Yamada *et al.*, 2016). However, this method was impractical for actual analyses, requiring large computational resources in both space and time to perform all-to-all pairwise alignments by dynamic programming (DP) (Needleman and Wunsch, 1970), which are used for a guide tree and a scoring function similar to COFFEE (Notredame *et al.*, 1998). Here, we introduce a scalable variant, G-large-INS-1, which has equivalent accuracy to G-INS-1 and is applicable to 50 000 or more sequences. Our strategies to reduce computational costs are (i) parallelization across multiple machines and/or processor cores using MPI and Pthreads to increase speed and (ii) the use of a high-speed shared filesystem, which is becoming common for processing big data. An MPI-based parallelization of another high-accuracy MSA method, MSAProbs, was recently released (González-Domínguez *et al.*, 2016), but it cannot be applied to thousands of sequences. The present update of MAFFT is designed to satisfy the need for accurately aligning large numbers of sequences but is not applicable to long genomic sequences since the length dependence of the computational cost is unchanged. The G-large-INS-1 option is available in MAFFT versions 7.355 or later and the online service (Katoh *et al.*, 2017).

Accuracy of G-large-INS-1 was compared with that of conventional G-INS-1 using different benchmarks, QuanTest (Le *et al.*, 2017) (Fig. 1a), HomFam (Sievers *et al.*, 2011), OXFam (Raghava *et al.*, 2003; Yamada *et al.*, 2016) and ContTest (Fox *et al.*, 2016) (Supplementary Table S1). Both methods ran with different input orders and/or minor variations in pairwise alignment and guide tree

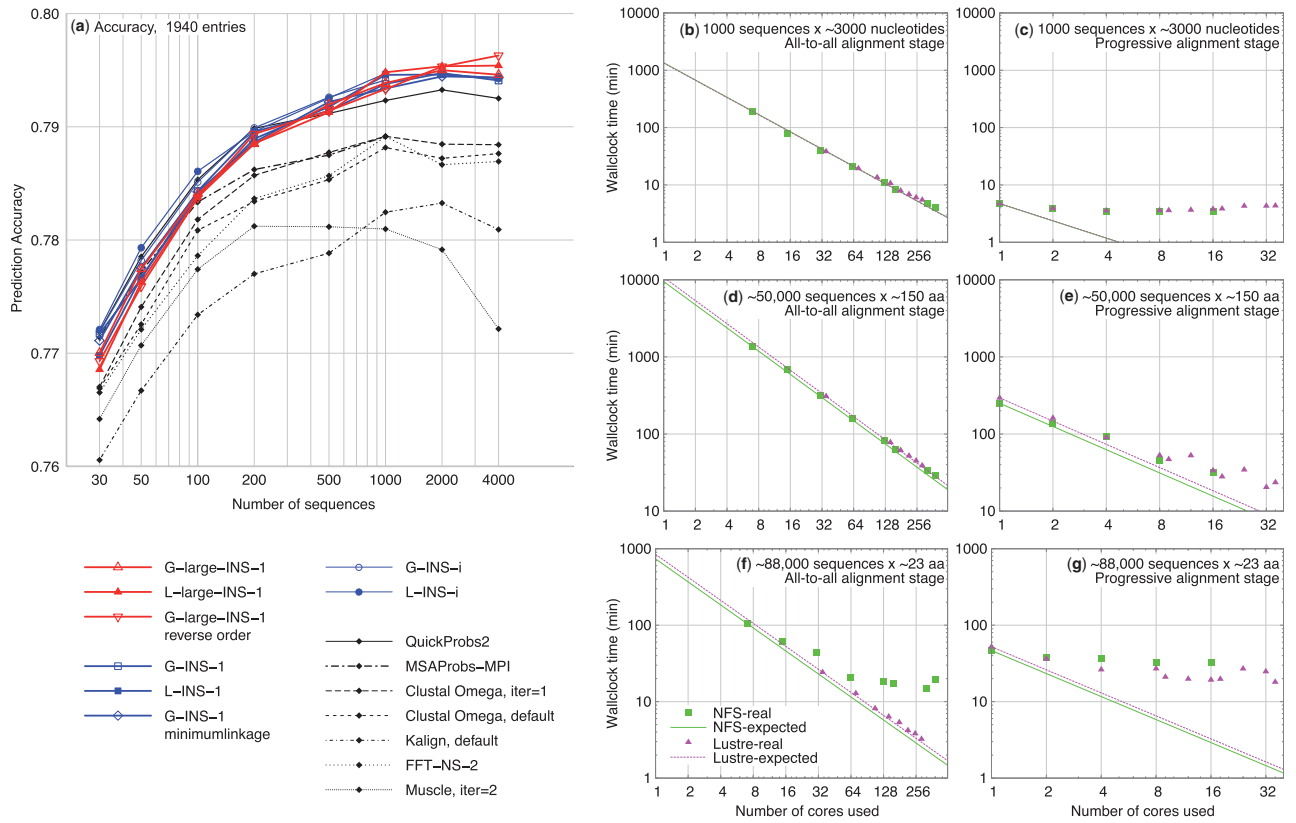

Fig. 1. (**a**) QuanTest. Accuracy of protein secondary structure prediction based on various sizes of MSAs by G-large-INS-1 (red bold lines), G-INS-1 (version 7.245; blue bold lines) and other popular methods. We used 1940 (out of 2265) entries so that JPred (Drozdetskiy *et al.*, 2015) can be consistently applied to the MSAs by all methods. (**b**)–(**g**), Parallelization efficiency of all-to-all alignment stage (**b**, **d** and **f**) and progressive stage (**c**, **e** and **g**) when applying G-large-INS-1 to LSU rRNA (**b**, **c**) sdr (**d**, **e**) and zf-CCHH (**f**, **g**). Green squares and magenta triangles are the computational time on NFS and Lustre filesystem, respectively. Lines are the expected time based on the cases using seven cores [NFS; green solid lines in (**b**), (**d**) and (**f**)], 35 cores [Lustre; magenta dotted lines in (**b**), (**d**) and (**f**)] and single core (**c**, **e** and **g**), assuming a perfect efficiency. The calculations with NFS (green) were performed on a heterogeneous cluster system (each node has 16–20 cores of Intel Xeon E5-2660 v3 2.6 GHz, E5-2680 2.7 GHz and E5-2670 v2 2.50 GHz with 64–128GB RAM). The calculations with the Lustre filesystem (magenta) were performed on Intel Xeon E5-2695 v4 2.10 GHz 36 cores with 256GB RAM per node using Lustre version 2.5.42

(see Supplementary Data) in order to assess instability of accuracy scores (Boyce *et al.*, 2015). In all cases, the difference between G-INS-1 (blue lines in Fig. 1a) and G-large-INS-1 (red lines) was small.

Large amounts of RAM are required if conventional tools for high-quality MSAs are applied to a large number of sequences. For example, MAFFT-L-INS-i and MSAProbs-MPI used at most 9.23GB and 74.8GB for a subset of 1000 sequences in QuanTest. For a larger subset (4000 sequences), MAFFT-G-INS-1 and QuickProbs2 (Gudyś and Deorowicz, 2017) used at most 26.0 GB and 411 GB RAM, respectively. In contrast, G-large-INS-1 used only 5.72GB at most, for the subset of 4000 sequences. Memory usage for larger problems (up to ~90 000 sequences) is shown in Supplementary Table S1, which suggests that this advantage increases with the number of sequences. Note that G-large-INS-1 uses files to save temporary data and thus requires a high-speed filesystem when the input sequences are very short, as discussed below.

Parallelization efficiency in three examples is shown in Figure 1(b–g), separately for two stages: (i) the all-to-all alignment stage (b, d and f) and (ii) the progressive alignment stage (c, e and g).

For LSU rRNA sequences (b, 1521–4102 bases, 1000 sequences randomly selected from the SEED alignment in Silva (Glöckner *et al.*, 2017) and protein sequences with usual lengths (d, 21–297 amino acids, 50 157 sequences, the 'sdr' family taken from HomFam), the wall-clock time for the all-to-all alignment stage decreased almost linearly with the number of cores used for the calculation. However, for a dataset with very short sequences (f, 12–35 amino acids, 88 345 sequences, the 'zf-CCHH' family taken from HomFam), the efficiency differs depending on filesystem: high in Lustre (shown with magenta triangles) but low in NFS (shown with green squares). This difference is due to the balance between calculation and disk operations. As noted earlier, a considerable amount of temporary data is written in parallel into the filesystem: approximately 218 MB, 100 GB and 142 GB for LSU rRNA, 'sdr' and 'zf-CCHH', respectively, in the examples shown here. Overhead due to these disk operations is almost negligible in the former two cases but not in the latter case, where alignment of ~23 amino acids takes only a short time in comparison with the time to write the temporary data to disk using NFS.

Figure 1c, e and g suggest that the wall-clock time of the progressive stage varies for each run and does not linearly decrease, but usually this is not a speed-limiting step. CPU time and wall-clock time for various problems are shown in Supplementary Table S1.

Until now, it was necessary to use highly approximate methods, such as the FFT-NS-2 option of MAFFT or the progressive option of Clustal Omega, in order to construct large MSAs. In terms of the MSA itself, the accuracy of these methods tends to decrease along with the increase in the number of sequences. This was first pointed out by Sievers *et al.* (2013) and confirmed by Le *et al.* (2017). The

increase in accuracy observed in Figure 1a for more than 200 sequences is due to the prediction phase not due to the alignment phase (see the last section in Supplementary Data and black dashed lines in Supplementary Fig. S1). As a result, it was difficult to know how many sequences should be included in an MSA. With more sequences, the MSA has richer comparative information, but the alignment quality is expected to decrease. The optimal balance between these two factors may differ by case. In contrast, the accuracy of G-large-INS-1 and G-INS-1 (red and blue dashed lines in Supplementary Fig. S1) was robust to data size in this test. The number of sequences to include in the MSA can now be determined simply based on the computational resources available and the requirements for the downstream analysis.

Acknowledgements

The authors thank Daron M. Standley and John Rozewicki, Osaka University, and Shun Sakuraba, the University of Tokyo, for discussion and computational support. The NIG supercomputer at ROIS National Institute of Genetics and the Reedbush System in the Information Technology Center, the University of Tokyo, were used.

Funding

This work was supported by JSPS KAKENHI [grant numbers JP16K07464 (to K.D.Y., K.T. and K.K.) and JP17J06457 (to T.N.)] and Platform Project for Supporting Drug Discovery and Life Science Research [grant numbers JP17am0101110 (to T.N., K.D.Y. and K.T.) and JP17am0101108 (to K.K.)] from AMED, Japan.

Conflict of Interest: none declared.

References

Boyce,K. *et al*. (2015) Instability in progressive multiple sequence alignment algorithms. *Algorithms Mol Biol*, **10**, 26.

Drozdetskiy,A. *et al*. (2015) JPred4: a protein secondary structure prediction server. *Nucleic Acids Res*., **43**, W389–W394.

Fox,G. *et al*. (2016) Using de novo protein structure predictions to measure the quality of very large multiple sequence alignments. *Bioinformatics*, **32**, 814–820.

Glöckner,F.O. *et al*. (2017) 25 years of serving the community with ribosomal RNA gene reference databases and tools. *J. Biotechnol*., **261**, 169–176.

González-Domínguez,J. *et al*. (2016) MSAProbs-MPI: parallel multiple sequence aligner for distributed-memory systems. *Bioinformatics*, **32**, 3826–3828.

Gudyś,A. and Deorowicz,S. (2017) QuickProbs 2: towards rapid construction of high-quality alignments of large protein families. *Sci. Rep*., **7**, 41553.

Katoh,K. *et al*. (2017) MAFFT online service: multiple sequence alignment, interactive sequence choice and visualization. *Brief Bioinformatics*, in press.

Le,Q. *et al*. (2017) Protein multiple sequence alignment benchmarking through secondary structure prediction. *Bioinformatics*, **33**, 1331–1337.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol*., **48**, 443–453.

Notredame,C. *et al*. (1998) COFFEE: an objective function for multiple sequence alignments. *Bioinformatics*, **14**, 407–422.

Raghava,G.P.S. *et al*. (2003) OXBench: a benchmark for evaluation of protein multiple sequence alignment accuracy. *BMC Bioinformatics*, **4**, 47.

Sievers,F. *et al*. (2011) Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol. Syst. Biol*., **7**, 539.

Sievers,F. *et al*. (2013) Making automated multiple alignments of very large numbers of protein sequences. *Bioinformatics*, **29**, 989–995.

Yamada,K.D. *et al*. (2016) Application of the MAFFT sequence alignment program to large data—reexamination of the usefulness of chained guide trees. *Bioinformatics*, **32**, 3246–3251.