Recent developments in the MAFFT multiple sequence alignment program

Kazutaka Katoh and Hiroyuki Toh

Submitted: 14th January 2008; Received (in revised form): 26th February 2008

Abstract
The accuracy and scalability of multiple sequence alignment (MSA) of DNAs and proteins have long been and are still important issues in bioinformatics. To rapidly construct a reasonable MSA, we developed the initial version of the MAFFT program in 2002. MSA software is now facing greater challenges in both scalability and accuracy than those of 5 years ago. As increasing amounts of sequence data are being generated by large-scale sequencing projects, scalability is now critical in many situations. The requirement of accuracy has also entered a new stage since the discovery of functional noncoding RNAs (ncRNAs); the secondary structure should be considered for constructing a high-quality alignment of distantly related ncRNAs. To deal with these problems, in 2007, we updated MAFFT to Version 6 with two new techniques: the PartTree algorithm and the Four-way consistency objective function. The former improved the scalability of progressive alignment and the latter improved the accuracy of ncRNA alignment. We review these and other techniques that MAFFT uses and suggest possible future directions of MSA software as a basis of comparative analyses. MAFFT is available at http://align.bmr.kyushu-u.ac.jp/mafft/software/.

Keywords: large-scale sequence alignment; fast tree building; ncRNA structural alignment; amino acid sequence alignment

INTRODUCTION
Multiple sequence alignment (MSA) is an important step in various types of comparative studies of biological sequences. MSA is used in phylogenetic inference, conserved region detection, structure prediction of noncoding RNAs (ncRNAs) and proteins and many other situations. For an easy MSA problem, such as an alignment consisting of a small number (<100) of short (<5000) sequences with global and high similarity (percent identity of >40% for protein cases and >70% for nucleotide cases), most of the current programs return a correct MSA, and no special consideration is needed. However, if all three of these conditions are not met, then the construction of an MSA can be a difficult task from both computational and biological viewpoints.

There is an established method based on the Dynamic Programming (DP) algorithm for calculating a pairwise alignment (an alignment between two sequences) [1–3] with a time complexity of \(O(L^2)\), where \(L\) is the sequence length. However, when more than two sequences must be aligned, the situation is somewhat complicated. Theoretically, the DP algorithm can be extended for cases of more than two sequences, but the time and space complexities of the naively extended algorithm, \(O(L^N)\), are impossibly large, where \(N\) is the number of sequences. Finding the exactly optimum MSA quickly becomes computationally intractable when the number of sequences increases [4]. Considerable efforts have been made to obtain the optimum MSA of \(~10\) sequences [5–10], which is still substantially smaller than the alignment size biologists now need. Therefore, some sort of heuristics are inevitable.

Even if the optimal MSA is successfully obtained, it is not always the correct solution from a biological viewpoint [11, 12]. This suggests that we should pay attention to a biologically relevant objective function, as well as to algorithmic techniques for obtaining the optimum solution. This is one of the reasons why various multiple sequence alignment schemes have...
been extensively studied to date, but there is no definitive one. Moreover, the accuracy of multiple alignment is improved by adding homologs or profiles [13–15]. This is because homologs make family-specific information available and enrich the profiles used in the multiple alignment processes [16]. Recent protein MSA studies indeed tended to use external sequence information [17–19]. Therefore, for an alignment program, the ability to handle many sequences is an important factor for yielding accurate results, as well as for large-scale analyses.

The MAFFT sequence aligner [20] was originally developed to perform the rapid calculation of an MSA consisting of a large number of sequences. A fast group-to-group alignment algorithm based on fast Fourier Transform (FFT) [20] and an approximate distance calculation method (the 6mer method) [20–23] facilitate the rapid calculation. Due to the increasing necessity for MSA of distant homologs, in 2005, we sought to improve the accuracy of MAFFT, and released Version 5 [14], which adopted a new objective function, the summation of a traditional weighted sum-of-pairs (WSP) score [24], and a consistency score similar to COFFEE [25] calculated from all-to-all pairwise alignments before constructing an MSA.

As a result, the current version of MAFFT has several options, as listed in Table 1, and covers various types of MSA problems, ranging from a small alignment consisting of distantly related sequences to a large-scale alignment. Recent benchmark studies under various conditions [26–30] consistently concluded that MAFFT is one of the best choices.

However, MAFFT does not completely cover all of the situations that biologists encounter. Especially for distantly related sequences, the use of multiple independent methods is important. The different MSAs computed by independent methods can be subjected to meta-aligners such as M-Coffee [31], to generate a more accurate MSA than those yielded by individual tools. The consensus among the different MSAs also provides information about which sites were reliably aligned [32, 33]. It should also be noted that different alignments sometimes result in quite different trees in phylogenetic analyses [34–36]. Such contradiction is partially (but not completely) avoided, by subjecting only reliably aligned sites to a phylogenetic inference.

In this article, after reviewing the general multiple alignment algorithms implemented in the MAFFT sequence aligner, we describe two new techniques introduced in Version 6: (i) a new tree-building algorithm, PartTree, for handling even larger numbers of sequences and (ii) a multiple ncRNA alignment framework incorporating structural information. We also describe some utility options that were added in Version 6 and provide tips to produce a reasonable alignment efficiently. For situations outside the scope of MAFFT, we introduce alternative tools developed by other groups.

**GENERAL ALGORITHMS**

**Terms and basic concepts**

**Sequence, alignment, homology and gap**

A sequence alignment is a set of corresponding residues among a collection of nucleotide or amino acid sequences. The sequences can be protein- or RNA-coding sequences or noncoding nucleotide sequences, such as introns or spacers. The sequences involved in an alignment are assumed to be homologous; that is, derived from a single common ancestral sequence. Aligned residues are usually interpreted as sharing their evolutionary origin. When a sequence has no corresponding residue because of an insertion or deletion event, the position is displayed as ‘--’ or another symbol and is called a ‘gap’. Most alignment programs do not attempt to filter out nonhomologous sequences, leaving the decision of what sequences to include in the MSA as an external decision for the user. However, this problem is sometimes important in actual analyses.

**Global homology and local homology**

Some MSA methods assume that all of the input sequences are globally alignable; that is, the entire regions of the sequences are assumed to be homologous, but this assumption does not necessarily agree with real analyses. Local alignment methods avoid the assumption of global homology. Some MSA methods, such as DIALIGN [37–39] and T-Coffee, have a facility to incorporate a local alignment algorithm to detect short patches of strong sequence similarity.

**Alignment of genomic sequences**

Unlike database-search programs [40, 41], most MSA programs try to include all of the residues in the input sequences, even when a local alignment algorithm is employed in a part of the calculation process. This policy makes the programs impractical when there are large nonhomologous regions within the sequences. We sometimes encounter such
Table 1: Options of MAFFT Version 6.5

<table>
<thead>
<tr>
<th>Option name</th>
<th>Command</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>For a large-scale alignment ((N &gt; \sim 10000)). Progressive methods with the PartTree algorithm</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NW-NS-PartTree</td>
<td>mafft --parttree --retree 1</td>
<td>Distance is by the 6mer method</td>
</tr>
<tr>
<td>NW-NS-PartTree2</td>
<td>mafft --parttree --retree 2</td>
<td>Distance is by the 6mer method. Guide tree is rebuilt</td>
</tr>
<tr>
<td>NW-NS-DPartTree</td>
<td>mafft --dparttree --retree 1</td>
<td>Distance is estimated based on DP</td>
</tr>
<tr>
<td>NW-NS-DPartTree2</td>
<td>mafft --dparttree --retree 2</td>
<td>Distance is estimated based on DP. Guide tree is re-built</td>
</tr>
<tr>
<td>NW-NS-FastaPartTree</td>
<td>mafft --fastaparttree --retree 1</td>
<td>Requires FASTA ([40]) to estimate distances</td>
</tr>
<tr>
<td>NW-NS-FastaPartTree2</td>
<td>mafft --fastaparttree --retree 2</td>
<td>Requires FASTA ([40]). Guide tree is rebuilt</td>
</tr>
<tr>
<td>For a medium-scale alignment ((-10000 &gt; N &gt; \sim 200)), Progressive methods</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FFT-NS-1</td>
<td>mafft --retree 1</td>
<td>Approximately two times faster than the default</td>
</tr>
<tr>
<td>FFT-NS-2</td>
<td>mafft</td>
<td>Default</td>
</tr>
<tr>
<td>For a small-scale alignment ((N &lt; \sim 200, L &lt; \sim 10000)), Iterative refinement methods</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FFT-NS-i</td>
<td>mafft-fftnsi</td>
<td>Fastest of the four in this category. Uses WSP score only</td>
</tr>
<tr>
<td>G-INS-i</td>
<td>mafft-ginsi</td>
<td>Uses WSP score and consistency score from local alignments</td>
</tr>
<tr>
<td>L-INS-i</td>
<td>mafft-linsi</td>
<td>Uses WSP score and consistency score from global alignments</td>
</tr>
<tr>
<td>E-INS-i</td>
<td>mafft-einsi</td>
<td>Uses WSP score and consistency score from local alignments with a generalized affine gap cost</td>
</tr>
<tr>
<td>For a small-scale RNA alignment ((N &lt; \sim 50, L &lt; \sim 10000)), Structural alignment methods</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q-INS-i</td>
<td>mafft-qinsi</td>
<td>Requires no external structural alignment programs</td>
</tr>
<tr>
<td>X-INS-i-scarnapair</td>
<td>mafft-xinsi --scarnapair</td>
<td>Requires MXSCARNA (Tabei et al., submitted for publication)</td>
</tr>
<tr>
<td>X-INS-i-larapair</td>
<td>mafft-xinsi --larapair</td>
<td>Requires LaRA ([78])</td>
</tr>
<tr>
<td>X-INS-i-foldalignlocalpair</td>
<td>mafft-xinsi --foldalignlocalpair</td>
<td>Requires FOLDALIGN ([79]), Uses the local alignment option</td>
</tr>
<tr>
<td>X-INS-i-foldalignglobalpair</td>
<td>mafft-xinsi --foldalignglobalpair</td>
<td>Requires FOLDALIGN ([97]), Uses the global alignment option</td>
</tr>
<tr>
<td>If not sure which option to use</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Automatic</td>
<td>mafft --auto</td>
<td>Selects an appropriate option from FFT-NS-2, FFT-NS-i and L-INS-i, according to the size of input data</td>
</tr>
</tbody>
</table>

\(N\) is the number of sequences and \(L\) is the sequence length.

situations when aligning genomic sequences. In such cases, MAFFT consumes a large amount of time. Instead, MLAGAN \([42]\) and MAVID \([43]\) are useful.

**Order of residues to be aligned**

When we are handling rearranged genomic sequences or mosaic proteins with rearranged multiple domains, the order of alignable residues can differ among the sequences. In such a case, tools without the assumption of the conservation of the order of aligned residues should be used. ABA \([44]\) (for protein/DNA), ProDA \([45]\) (for protein), TBA \([46]\) (for genomic DNA) and MAUVE \([47]\) (for genomic DNA) are available.

**Progressive method—FFT-NS-2**

The progressive method \([48, 49]\) is the most commonly used multiple alignment algorithm. Clustal W \([50, 51]\), MAFFT, POA \([52]\), Kalign \([53]\) and many other MSA packages use this method with various modifications. The procedure of the progressive method implemented in MAFFT is schematically illustrated in Figure 1A. A guide tree, a tentative tree only used for constructing an alignment, is created based on all-to-all pairwise comparisons, and an MSA is constructed using a group-to-group alignment algorithm at each node of the guide tree.

To achieve a reasonable balance between speed and accuracy, MAFFT \([20]\) adopts, by default, a two-cycle progressive method, called FFT-NS-2, in which low-quality all pairwise distances are rapidly calculated, a tentative MSA is constructed, refined distances are calculated from the MSA, and then the second progressive alignment is performed, as shown in Figure 1A. In addition, MAFFT uses two key techniques, an FFT-based group-to-group alignment algorithm \([20]\) (Figure 2) and the 6mer method \([20–23]\) for all pairwise comparisons, to reduce the CPU time of progressive methods. The time complexity of the progressive method implemented in MAFFT is basically \(O(N^2L) + O(NL^2)\), where \(L\) is the sequence length and \(N\) is the number of sequences. The first term corresponds to the guide tree calculation and the second term corresponds to the group-to-group alignment stage. When the input sequences are highly similar to each other, it is reduced to \(O(N^2L) + O(NL) = O(N^2L)\), because of the FFT-based alignment method (See \([20]\) for details).
Its space complexity is basically \(O(N^2) + O(L^2) + O(NL)\). When the sequence length exceeds the threshold (set as 10,000 residues at present), FFT-NS-2 automatically switches the DP algorithm to a memory saving one [54] and the space complexity becomes \(O(N^2) + O(NL)\). On a current desktop computer, this method can be applied to an MSA consisting of up to \(\sim 10,000\) sequences. The maximum length depends on the similarity level: \(\sim 10,000\) residues for distantly related sequences or \(\sim 500,000\) residues for closely related sequences with global homology.

The progressive method has a drawback in that once a gap is incorrectly introduced at a step, the gap is never removed in later steps. To overcome this drawback, there are two types of solutions, the iterative refinement method [55–61] and the consistency-based method [25, 62–64]. These two procedures are quite different: the former tries to correct mistakes in the initial alignment, whereas the latter tries to avoid mistakes in advance, but both work well to improve the alignment accuracy.

**Iterative refinement method with the WSP score—FFT-NS-i**

In the iterative refinement method, an objective function that represents the ‘goodness’ of the MSA is explicitly defined. An initial MSA, calculated by the progressive or another method, is subjected to an iterative process and is gradually modified so that the objective function is maximized, as shown in Figure 1B. Various combinations of objective functions and optimization strategies have been proposed to date [55–61]. Among them, Gotoh’s iterative refinement method, PRRN [16], is the most successful one, and it forms the basis of recent methods, including MAFFT, MUSCLE [23, 65] and PRIME [66]. The iterative alignment option of MAFFT, called FFT–NS–i, uses the weighted sum-of-pairs (WSP) objective function [24]. As shown in Figure 1B, an MSA is partitioned into two groups, which are then realigned using an approximate group-to-group alignment algorithm [20]. The new MSA replaces the old one if it has a higher score. This process is repeated until no more improvements
are made. To save computation time, the partitions of the MSA are restricted to those corresponding to the branches of a tree among the sequences [67]. The time complexity of this method is \( O(N^2L) + O(NL^2) \) and the space complexity is \( O(N^2) + O(L^2) + O(NL) \) or \( O(N^2) + O(NL) \), depending on the sequence length, as in the case of the progressive option. On a current desktop computer, this method can be applied to an MSA consisting of up to \( \sim 500 \) sequences. The maximum length depends on the similarity level: \( \sim 10,000 \) residues for distantly related sequences or \( \sim 500,000 \) residues for closely related sequences with global homology.

**Iterative refinement method with consistency and WSP scores—G-INS-i**

T-Coffee [64], ProbCons [68] and other methods [69, 70] take an entirely different approach to overcome the drawback of the progressive method by using a consistency criterion, in which an MSA consistent with pairwise alignments is judged to be relevant. Several types of consistency criteria were described previously [25, 62, 63], but T-Coffee achieved a great improvement in accuracy by using the COFFEE criterion [25] together with the library extension technique [64] in the progressive method. In 2005, MAFFT adopted a consistency criterion into the iterative refinement method [14]. Instead of the library extension procedure, which plays an important role in T-Coffee but requires considerable computing power, we took a iterative strategy with an objective function of the summation of the WSP score [24] and a consistency-based score like COFFEE [25]. This method was implemented as the G-INS-i option. Its time complexity is \( O(N^2L^2) \). Its space complexity is at least \( O(N^2) + O(L^2) + O(NL) \) but greatly depends on the similarity level. The G-INS-i option assumes that the input sequences are globally alignable. This option is suitable when the lengths of the input sequences are similar to each other, as in Figure 3A. On a current desktop computer, this method can be applied to an MSA consisting of up to \( \sim 200 \) sequences. The maximum length is \( \sim 5000 \) residues.

**NEW FEATURES IN VERSION 6**

**Variants of G-INS-i—L-INS-i and E-INS-i for large gaps**

After the publication of MAFFT Version 5 [20], we added the L-INS-i and E-INS-i options, which are variants of G-INS-i, to the MAFFT package. Their basic procedures are the same as that of G-INS-i, but different algorithms are used in the pairwise alignment stage. L-INS-i uses a local pairwise alignment [3] with the affine gap cost [2], while
E-INS-i uses a local pairwise alignment with the generalized affine gap cost [71], in which the unalignable region is left unaligned at the pairwise alignment stage. L-INS-i and E-INS-i can be applied to the cases where alignments those in Figure 3B–C are expected, respectively. Note that E-INS-i also includes all of the residues in the alignment during the rest of its procedure, and the resulting alignment is always a full-length alignment. Moreover, it is better to see how MSA programs work and to try some independent methods with various parameters, particularly when a set of distantly related sequences is aligned [33, 72]. T-Coffee, ProbAlign and DIALIGN can be alternative choices for such situations requiring large gaps.

PartTree
Since increasing amounts of sequence data are being generated from large-scale sequencing projects, scalability is now critical in many situations. As noted above, the time complexity of the progressive method is \( O(N^2L) + O(NL^2) \). The first term corresponds to all-to-all comparisons of input sequences and guide tree building by the UPGMA method [23, 73]. As this term can be the time-limiting factor when large numbers (10 000 or more) of sequences are aligned, it is desirable to omit the two steps. Without a guide tree, however, the resulting alignment highly depends on the input order, and the quality is not acceptable in most cases.

Hence, we developed a scalable tree-building algorithm, PartTree [74], to generate a guide tree from a set of unaligned sequences with a time complexity of \( O(N \log N) \). PartTree is a divisive clustering algorithm. In summary, \( n \) representative sequences are randomly selected from the input sequences and then the other sequences are grouped with the \( n \) representatives, according to the similarity. The calculation of similarity is performed only \( nN \) times at this time. The UPGMA tree among \( n \) representatives is calculated. This step is recursively repeated for each of the \( n \) groups, unless the group has only one sequence. The \( n \) UPGMA trees returned by child processes are combined into a single tree. In total, the similarity calculation is performed \( N \log N \) times, on average. Thus, this algorithm is faster than the conventional UPGMA algorithm, which requires all pairwise similarity calculations with a time complexity of \( O(N^2) \).

The PartTree option implemented in MAFFT Version 6 (Table 1) can successfully align a dataset consisting of a large number (~60 000) of homologous sequences, at the cost of an accuracy loss of ~2%. Various combinations of distance estimation methods (FASTA-based, DP-based or 6mer-based), a parameter \( n \) (the number of representatives) and the number of re-estimations of the guide tree (as shown in Figure 1A) can be selected, according to the needs for balance between accuracy and speed. See the original paper [74] for the benchmark results.

RNA alignment
The importance of RNA alignment is increasing, since the discovery of functional ncRNAs. MAFFT Version 6 has two new options, Q-INS-i and X-INS-i, for RNA alignment. Both methods consider secondary structure information, as a form of base-pairing probability, predicted by either the McCaskill algorithm [75] or the CONTRAfold algorithm [76]. In Q-INS-i, the base-pairing probability is incorporated into the resulting alignment with a new objective function, Four-way Consistency (Katoh and Toh, submitted). In X-INS-i, the structural information is also used.

---

**Figure 3:** Globally alignable (A), locally alignable (B), and long internal gaps (C). '-' represents a gap, 'X' represents an aligned residue and 'o' is an unalignable residue.
at the pairwise alignment stage, in addition to the Four-way Consistency objective function. At present, three different external structural alignment programs, MXSCRANA [77], LaRA [78] and FOLDALIGN [79], are supported as the source of pairwise structural alignments for X-INS-i. Although MXSCRANA and LaRA are multiple RNA alignment programs themselves, only their pairwise alignment functions are used.

A benchmark result (Table 2) indicates that the performances of structural alignment methods for multiple RNAs have rapidly improved in 2007, as a result of intensive studies by many groups [78, 80–88], and that the combination of X-INS-i and SCARNA (denoted as X-INS-i-scarnapair in Table 2) is the most accurate method, according to most benchmark criteria. Moreover, the calculation time of X-INS-i-scarnapair is shorter than those of other accurate methods, such as RNA Sampler [84], MASTR [87] and Murlet [83]. The difference in accuracy between X-INS-i-scarnapair and MXSCRANA reflects the improvement gained by the X-INS-i framework, because these two methods use the same pairwise structural alignment algorithm, SCARNA [77].

### Group-to-group alignment and seed alignment

MAFFT Version 6 has the mafft-profile program, which functions like the profile alignment option of Clustal W. When two alignments are given, the mafft-profile program converts each alignment into a profile and returns an alignment between the two alignments.

```
% mafft-profile aligned_group1 aligned_group2 > output
```

This is sometimes useful in actual analyses, but needs consideration of the phylogenetic relationship between the two groups. The profile alignment assumes that the two input alignments are phylogenetically separated, as in Figure 4A or B.

When another phylogenetic relationship is expected, as in Figure 4C, the profile alignment could introduce misalignments. In fact, we sometimes encounter such a situation when we want to add
a newly determined sequence into an established alignment, which was already adjusted by eye or taken from an annotated database. For such a case, MAFFT Version 6 provides another type of group-to-group alignment option based on the consistency criterion,

```bash
% mafft-linsi --seed aligned_sequences new_sequence > output
```

which discards all of the gaps and then makes a new alignment consisting of all of the members of `aligned_sequences` and `new_sequence`, in which the alignment within `aligned_sequences` is exactly reconstructed. Thus, `new_sequence` is first aligned with the nearest sequence (marked with *) in `aligned_sequences` and then is aligned with the other members. This can be applied to a situation like that in Figure 4C.

The seed alignment option can be used in a more complex situation like that in Figure 4D, in which we already have a skeleton alignment, based on structural information or manual annotation, and we have to add multiple unaligned homologs into the alignment. In such a case,

```bash
% mafft-linsi --seed aligned_sequences unaligned_sequences > output
```

makes an entire alignment while preserving the skeleton alignment.

**AVAILABILITY**

The source code of MAFFT Version 6 is available at [http://align.bmr.kyushu-u.ac.jp/mafft/software/](http://align.bmr.kyushu-u.ac.jp/mafft/software/). The code for McCaskill routine was taken from the Vienna RNA package Version 1.5 [89] and the McCaskill-MEA package [90]. The binaries for Macintosh, Windows and Linux are also available at the same site. We provide alignment and phylogenetic inference services (Figure 5) at [http://align.bmr.kyushu-u.ac.jp/mafft/online/server/](http://align.bmr.kyushu-u.ac.jp/mafft/online/server/). For the phylogenetic inference, users can chose either the NJ [91] or UPGMA [73] method. For the NJ method, several methods for distance estimation can be selected: the Poisson correction, the maximum-likelihood (ML) estimation assuming the JTT [22] or WAG model [92] for protein alignment; and the Jukes-Cantor correction [93] for nucleotide alignment. We modified the MOLPHY package [94] to consider the variable substitution rate across sites with the discrete Γ model [95] and use it in the ML distance estimation from a protein alignment. Bootstrap analysis is also supported. Alignments and phylogenetic trees are visualized with the Jalview [96] and ATV [97] viewers, respectively.
FUTURE DIRECTIONS

Consideration of RNA structure

As of December 2007, X-INS-i-scarnapair is one of the most accurate methods for multiple structural RNA alignment. With the current implementation with a time complexity of $O(N^2L^3)$, X-INS-i-scarnapair is already faster than other accurate methods, such as RNA Sampler [84], MASTR [87] and Murlet [83], but the time complexity of X-INS-i-scarnapair can be further reduced to $O(NL^3) + O(N^2L^2)$ if the SCARNA source becomes open [see Katoh and Toh, submitted for publication for details].

Many research groups are now working on the RNA alignment issue [77–88], and the accuracy and speed of ncRNA aligners have rapidly improved in the last several months, as shown in Table 2. Many of them are based on the Sankoff algorithm [98], which simultaneously performs alignment and secondary structure prediction with a time complexity of $O(L^{3N})$. For pairwise structural alignment ($N=2$), several successful methods are becoming available [77, 79], which reduced the time complexity from $O(L^6)$ to $O(L^3)$ or so, by introducing various approximations. However, it does not seem fruitful to directly extend the Sankoff algorithm to multiple alignment, for the reason explained in the Introduction section, as well as the problem of time complexity.

This is one of the motivations behind the development of the X-INS-i framework for multiple structural RNA alignment based on the Four-way Consistency (Katoh and Toh, submitted for publication). We are ready to support any pairwise structural alignment algorithm, regardless of whether it is Sankoff-based or not, to be extended to the multiple alignment problem using our framework, which was designed to accept various types of pairwise structural alignments and combine them into a single multiple structural alignment.

Consideration of protein structure

The consideration of structural information is also important for protein alignment, and thus many efforts have been made. SPEM [17], MUMMALS [70], PROMALS [18] and other methods incorporate predicted structural information into an alignment, like the RNA aligners noted above. In contrast, 3DCoffee [99], Expresso [100] and other methods incorporate experimentally determined protein structure information into an alignment by using external structural alignment algorithms, such
as SAP [101], and structure–sequence alignment algorithms, such as FUGUE [102]. Both of these approaches achieved considerable improvement in the accuracy [33]. The latter approach seems promising, because many protein structures are being determined along with the progress of structural genomics. We are planning to explore a combination of MAFFT with the ASH [103–105] structural alignment algorithm.

**Scalability**
The FFT-based alignment algorithm, the PartTree algorithm and other techniques successfully improved the scalability of MSA. However, these are only applicable to a progressive method, FFT-NS-2, but not to the most accurate methods, G-INS-i, L-INS-i and E-INS-i. The maximum size of the sequence data for these options is currently ~200 sequences × ~5000 sites or so, on a typical desktop computer. We are planning to parallelize the pairwise alignment part of these three options. The maximum data size can be extended by parallelization, although the order of time complexity does not change.

**Determining an appropriate set of sequences and positions to be included within an MSA**
For phylogenetic analyses, we sometimes encounter a serious problem, in terms of which sequences should be included in an MSA and a phylogenetic tree, and which sequences should be excluded. As a large number of homologs are available from databases, such a problem becomes quite bothersome. There are several types of unusual sequences that degrade the accuracies of alignment and phylogenetic inference: (i) fragment sequences, (ii) amino acid sequences incorrectly translated from genomic data and (iii) nonhomologous sequences, etc.

Gouveia-Oliveira et al. [106] recently described a tool, MaxAlign, that deletes unusual sequences from a given MSA to maximize the size of ‘alignment area’, the number of residues in gap-free columns. MaxAlign seems to be an interesting approach and it may be more useful if an MSA method itself automatically determines the sequences to be included within the alignment. One possible way is to extract a commonly aligned region by all-to-all pairwise local alignments. However, such a method may miss a considerable part of alignable residues, because pairwise alignment is usually less sensitive than multiple alignment. A iterative application of multiple alignment and MaxAlign may be worth trying.

There can be different situations where unusual sequences should not be excluded. For example, an MSA itself is useful to identify misidentified genes and other unusual sequences. Therefore, an alignment algorithm that is robust to unusual data is also an important issue.

**Incorporation and extraction of biological knowledge in an MSA**
In order to construct a biologically relevant MSA, we have to consider the structural, functional and evolutionary information, as well as the optimality with respect to a given scoring system. Manual inspection based on biological knowledge will thus remain important [35], although it is becoming difficult with the increasing number of available sequences. In such a situation, the use of databases of annotated alignments [107, 108] will be a fruitful and practical way to construct an accurate MSA, as well as to extract solid information from an MSA [109]. Hence, more flexible frameworks and tools to build an MSA combining various types of alignment-related data, including structural alignments and manually annotated information, will become important, in addition to more relevant objective functions and faster algorithms.

**Key Points**
- MAFFT Version 6 has two major new features, the PartTree algorithm for handling a large number (~10 000) of sequences and the Four-way Consistency objective function for multiple structural alignment of ncRNAs.
- PartTree is a divisive recursive clustering algorithm with a time complexity of O(N log N). It is more scalable than the conventional UPGMA algorithm with a time complexity of O(N^2). The PartTree option can create a large alignment composed of ~60 000 sequences, at the cost of an accuracy loss of ~2%.
- The X-INS-i-scarnapair, which is a combination of an external pairwise structural RNA alignment method, SCARNA, and the Four-way Consistency objective function, is one of the most accurate methods for multiple RNA structural alignment. It requires less CPU time than other accurate structural alignment methods, such as RNA Sampler, MASTR and Murlet.
- Two different types of group-to-group alignment methods, the profile alignment option and the seed option, were implemented, in order to deal with the various possible phylogenetic relationships between two groups.
- MAFFT Version 6 has L-INS-i and E-INS-i options, which are variants of G-INS-i, the iterative refinement method with WSP and consistency scores. L-INS-i allows large terminal gaps, while E-INS-i is applicable to a dataset with internal unalignable regions.
Acknowledgements

We thank Hiroshi Suga, Kei-ichi Kuma and Go Sasaki for providing a part of the tree inference programs, which were developed at Takashi Miyata’s laboratory, Kyoto University, during 1991–2004. We also thank Kazuharu Misawa for a key idea on the FFT-based alignment algorithm. This work was supported by a Grant-in-Aid for ‘Comparative Genomics’ from MEXT of Japan.

References