Syst. Biol. 56(4):564–577, 2007

ISSN: 1063-5157 print / 1076-836X online
DOI: 10.1080/10635150701472164

Improvement of Phylogenies after Removing Divergent and Ambiguously Aligned Blocks from Protein Sequence Alignments

GERARD TALAVERA AND JOSE CASTRESANA

Department of Physiology and Molecular Biodiversity, Institute of Molecular Biology of Barcelona, CSIC, Jordi Girona 18, 08034 Barcelona, Spain; E-mail: jcvagr@ibmb.csic.es (J.C.)

Abstract.—Alignment quality may have as much impact on phylogenetic reconstruction as the phylogenetic methods used. Not only the alignment algorithm, but also the method used to deal with the most problematic alignment regions, may have a critical effect on the final tree. Although some authors remove such problematic regions, either manually or using automatic methods, in order to improve phylogenetic performance, others prefer to keep such regions to avoid losing any information. Our aim in the present work was to examine whether phylogenetic reconstruction improves after alignment cleaning or not. Using simulated protein alignments with gaps, we tested the relative performance in diverse phylogenetic analyses of the whole alignments versus the alignments with problematic regions removed with our previously developed Gblocks program. We also tested the performance of more or less stringent conditions in the selection of blocks. Alignments constructed with different alignment methods (ClustalW, Mafft, and Probcons) were used to estimate phylogenetic trees by maximum likelihood, neighbor joining, and parsimony. We show that, in most alignment conditions, and for alignments that are not too short, removal of blocks leads to better trees. That is, despite losing some information, there is an increase in the actual phylogenetic signal. Overall, the best trees are obtained by maximum-likelihood reconstruction of alignments cleaned by Gblocks. In general, a relaxed selection of blocks is better for short alignment, whereas a stringent selection is more adequate for longer ones. Finally, we show that cleaned alignments produce better topologies although, paradoxically, with lower bootstrap. This indicates that divergent and problematic alignment regions may lead, when present, to apparently better supported although, in fact, more biased topologies. [Bootstrap support; Gblocks; phylogeny; sequence alignment.]

Methods for the simultaneous generation of multiple alignments and phylogenetic trees are actively being pursued (Fleissner et al., 2005; Lunter et al., 2005; Redelings and Suchard, 2005; Wheeler, 2001), but, at present, common practice of phylogenetic analysis requires, as a first step, the generation of a multiple alignment of the sequences to be analyzed. It has been repeatedly shown that the quality of the alignment may have an enormous impact on the final phylogenetic tree (Kjer, 1995; Morrison and Ellis, 1997; Ogden and Rosenberg, 2006; Smythe et al., 2006; Xia et al., 2003). This is particularly true when sequences compared are very divergent and of different length, which makes necessary the introduction of gaps in the alignments.

Due to the computational requirements of optimal algorithms for multiple sequence alignments, different heuristic strategies have been proposed. The most widely used approach has been the progressive method of alignment (Feng and Doolittle, 1987) that, together with enhancements related to the introduction of gap penalties, was implemented in ClustalW (Thompson et al., 1994). In progressive methods, an initial dendrogram generated from the pairwise comparisons of the sequences is used to recursively build the multiple alignment, using dynamic programming (Needleman and Wunsch, 1970) in the last step. Dynamic programming is an exact algorithm that assures the best possible alignments for given gap penalties but, due to heavy computational requirements, it is only used for pairs of sequences or pairs of clades of the dendrogram and not for the whole multiple alignment. Several other heuristic multiple alignment methods have been recently introduced. They include T-Coffee (Notredame et al., 2000), Mafft (Katoh et al., 2005; Katoh et al., 2002), Muscle (Edgar, 2004), Probcons (Do et al., 2005), and Kalign (Lassmann and Sonnhammer, 2005), among others. All of them are based on the progressive method but include several iterative refinements to construct the final multiple alignment. The latter methods have been shown to outperform purely progressive methods in terms of alignment accuracy and, some of them, even in computational time. However, it has not been shown whether the greater alignment accuracy of more sophisticated methods leads to a significant improvement in phylogenetic reconstruction.

Proteins have some regions that, due to their functional or structural importance, are very well conserved, whereas other regions evolve faster both in terms of nucleotide substitutions and insertions or deletions (Henikoff and Henikoff, 1994; Herrmann et al., 1996; Pesole et al., 1992). That is, evolutionary rate heterogeneity affects to whole regions in addition to single positions. This type of regional rate heterogeneity is very challenging for phylogenetic reconstruction, not only in terms of homoplasy due to saturation (Yang, 1998), but also in terms of errors in homology during alignment.

Dealing with regions of problematic alignment is a matter of active debate in phylogenetics. Although some authors consider that it is best to remove such regions before the tree analysis (Castresana, 2000; Grundy and Naylor, 1999; Löytynoja and Milinkovitch, 2001; Rodrigo et al., 1994; Swofford et al., 1996), others think that there is an important loss of information upon removal of any fragment of the sequences already obtained (Aagesen, 2004; Lee, 2001) and that this practice should only be used as the last resource (Gatesy et al., 1993). A third, intermediate option, is the recoding of such regions using different strategies (Geiger, 2002; Lutzoni et al., 2000; Young and Healy, 2003), which allows the use of at least part of the information. Although these coded characters are most commonly analyzed with parsimony, it is

also possible to use them as independent partitions in Bayesian or likelihood frameworks.

In the present work we test, by using simulated protein alignments with gaps, which are the best alignment strategies for optimal phylogenetic reconstruction. Two preliminary considerations are necessary here. First, simulations of sequences may not cover all the complexity of evolution but have the advantage over real sequences that we know the tree from which they have been generated. There are some alignment sets curated from structural information that can be used to test alignment accuracy (Thompson et al., 2005), but the phylogenetic tree is unknown in these sets, thus making problematic their use for proving phylogenetic accuracy. Second, we have been working with simulated sequences that try to reflect the evolutionary patterns of proteins, and thus many of the conclusions extracted from our work cannot be directly extrapolated to other markers such as rRNA, which show very different evolutionary constraints (Gutell et al., 1994; Kjer, 1995; Xia et al., 2003).

In our analysis we used different alignment strategies of the simulated sequences to test if they make any difference in the final phylogenetic tree. We have selected ClustalW as the currently most used progressive alignment method (Thompson et al., 1994) and Mafft (Katoh et al., 2005) and Probcons (Do et al., 2005) as examples of more recently developed methods that have been shown to obtain very high scores in terms of alignment accuracy (Blackshields et al., 2006; Nuin et al., 2006). Simultaneously with the performance of the alignment programs, we tested whether removing blocks of problematic alignment actually leads to more accurate trees. We used for this purpose our previously developed Gblocks program (Castresana, 2000), which selects blocks following a reproducible set of conditions. Briefly, selected blocks must be free from large segments of contiguous nonconserved positions, and flanking positions must be highly conserved to ensure alignment accuracy. Several parameters can be modified to make the selection of blocks more or less stringent. Phylogenetic trees made by maximum likelihood (ML), neighbor joining (NJ), and parsimony of the reconstructed alignments show that, in almost all conditions tested, and at least for alignments that are not too short, the elimination of problematic regions by Gblocks leads to significantly better phylogenetic trees.

Materials and Methods

We simulated protein sequences by means of Rose (Stoye et al., 1998). This program allows the simulation of different substitution rates in different positions with a predetermined spatial pattern. This is a very important feature for testing the behavior of a program like Gblocks, which selects from alignments blocks of contiguous conserved positions with few nonconserved positions inside. This is the reason why a program that simulates among-site rate heterogeneity, but not regional heterogeneity, would not be valid to test the behavior of Gblocks. Thus, an important preliminary step in our simulations was the selection from real proteins of spatial patterns of site rates in order to use these parameters with Rose.

Selection of Evolutionary Rate Patterns

We extracted patterns of rate heterogeneity from real protein alignments using the program TreePuzzle (Strimmer and von Haeseler, 1996) with a model of among-site rate heterogeneity that assumed a Gamma distribution of rates. This distribution was approximated with 16 rate categories, which is the maximum number allowed in TreePuzzle. In particular, we took, from each position, the category and associated relative rate that contributed the most to the likelihood. Positions with rates >1 receive more mutations than the average and positions with rates <1 receive fewer mutations. This list of relative rates (whose average should be 1) were given to Rose to simulate different positions with different rates, creating conserved and divergent regions with lengths and boundaries that approximated those of a real protein. Proteins for extracting rate patterns were NAD2 and NAD4 (subunits 2 and 4 of the mitochondrial NADH dehydrogenase) from several metazoans (Castresana et al., 1998b), and COG0285 from the COG database, which includes mainly bacterial sequences (Tatusov et al., 2003). The three selected profiles produced similar conclusions regarding the best block selection strategy, and we used the NAD2 pattern to perform most of the tests. This pattern contained 361 positions but, after the introduction of further gaps by the simulation algorithm, the final simulated alignments reached approximately 400 positions. In order to simulate alignments of different length, independent simulations obtained with this pattern were concatenated 1, 2, 3, 4, and 8 times to generate final alignments of, approximately, 400, 800, 1200, 1600, and 3200 positions, respectively. The PAM evolutionary model (Dayhoff et al., 1978) was used to simulate the evolution of amino acids.

Selection of Phylogenetic Trees

Simulations with Rose were performed along phylogenetic trees of 16 tips with three different topologies, a purely asymmetric tree (Fig. 1a), an intermediate tree (Fig. 1b), and a symmetric tree (Fig. 1c). These known trees or "real trees" were manually constructed. The average and maximum length from the root to the tips was, for the asymmetric tree, 0.89 and 1.30 substitutions/position, respectively. The other trees had very similar values. The branch lengths of the three trees in Figure 1 were multiplied by factors of 0.5, 1, and 2, respectively, so that we used in total 9 phylogenetic trees. These trees had several short internal branches that made them difficult to resolve; thus, they are trees where the alignment strategy as well as the phylogenetic algorithm used were differentially effective. Simpler trees in terms of longer internodes were easily and equally reproduced by all methods and were not used here. Similarly, trees with a total smaller divergence tended to produce conserved alignments where the alignment method was not an issue and also not used here. Finally, these trees did

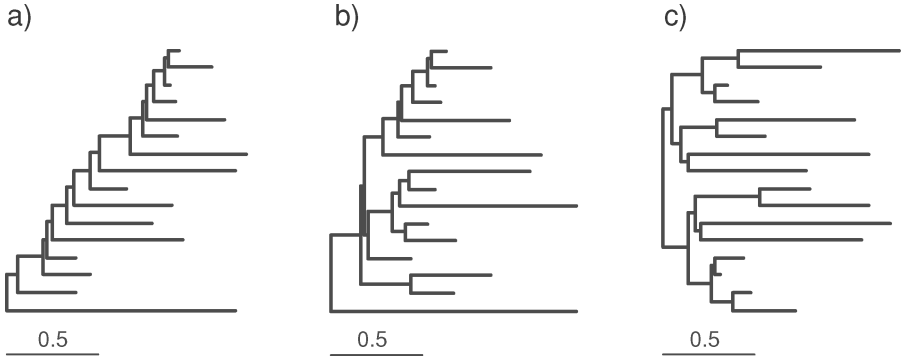

FIGURE 1. Asymmetric (a), intermediate (b), and symmetric (c) trees used in the simulations. The scale bar, in substitutions/position, corresponds to the trees with a divergence ×1.

not contain many closely related sequences, since we wanted to specifically measure differences in reproducing the overall shape of the tree and not differences in recovering the relationships among close sequences.

Gaps Introduced during the Simulations

The Rose program does not have any specific model for the introduction of gaps along the alignment. Rather, gaps are introduced with equal probability in all positions with a relative rate ≥1 (Stoye et al., 1998), which is a limitation of this program. To try to overcome this limitation, we used two different gap strategies within Rose. First, we used a single gap threshold for the whole alignment. After several trials, we considered a threshold of 0.0007 as a reasonable one for the divergence levels we analyzed, as deduced from visual inspection of the alignment (that is, eyeing that blocks of divergence and conservation were not so different from the real proteins used to construct the rate profiles). Even so, this threshold tended to produce too many gaps in conserved regions (not shown). In addition, we also generated alignments with two different gap thresholds, 0.001 and 0.0001, which we associated, respectively, to divergent and to conserved regions of the profiles. For doing so, we divided the rate profiles in blocks of homogeneous divergence (that is, each block was either mostly conserved or mostly divergent, which resulted in around 10 to 20 blocks for the different profiles). Then, we did the simulations for each block separately, and with its own gap threshold (high for divergent blocks and low for more conserved blocks). Finally, the different simulated blocks were concatenated. The phylogenetic results were similar with both gap strategies, but we mostly worked with simulations that had the two different gap thresholds, which we considered more realistic. In all cases we chose a vector of indels of the form [0.5, 0.4, 0.3, 0.2, 0.1], which reflects the relative frequency of indels with lengths from 1 to 5 amino acids, respectively.

Realignments of Simulated Sequences

Alignments generated by Rose were cleaned from gaps and new alignments were reconstructed using ClustalW version 1.83 (Thompson et al., 1994), Mafft version 5.531 (Katoh et al., 2002, 2005), and Probcons version 1.1 (Do et al., 2005). Default parameters were used in ClustalW and Probcons. All defaults were also used in Mafft except that a neighbor joining instead of a UPGMA tree was used as guide tree (option –nj). Alignments were cleaned from problematic alignment blocks using Gblocks 0.91 (Castresana, 2000), for which two different parameter sets were used. In one of them, which we call here stringent selection, and which is the default one in Gblocks 0.91, "Minimum Number of Sequences for a Conserved Position" was 9, "Minimum Number of Sequences for a Flank Position" was 13, "Maximum Number of Contiguous Nonconserved Positions" was 8, "Minimum Length of a Block" was 10, and "Allowed Gap Positions" was "None". In the second set, which we call relaxed selection, we changed "Minimum Number of Sequences for a Flank Position" to 9, "Maximum Number of Contiguous Nonconserved Positions" to 10, "Minimum Length of a Block" to 5, and "Allowed Gap Positions" to "With Half". The latter option allows the selection of positions with gaps when they are present in less than half of the sequences.

Original simulated alignments and Mafft realignments for 30 example simulations (the first five simulations generated with the symmetric and asymmetric trees) are provided as supplementary information (available online at http://systematicbiology.org).

Phylogenetic Reconstruction

Phylogenetic trees from the complete and the two different Gblocks alignments were estimated by ML, NJ, and parsimony. For ML trees we used the Phyml program version 2.4.4 (Guindon and Gascuel, 2003), with the Jones-Taylor-Thornton model of protein evolution (Jones et al., 1992) and four rate categories in the Gamma distribution. The Gamma distribution parameter and the proportion of invariable sites were estimated by the program. For NJ trees we used Protdist of the Phylip package version 3.63 (Felsenstein, 1989) with the Jones-Taylor-Thornton model to calculate pairwise protein distances, and Neighbor of the same package to calculate the NJ tree. For parsimony we used Protpars of the Phylip

package (Felsenstein, 1989) with 50 random initializations to ensure a thorough tree search. If no parsimony tree was obtained, which occurred in less than 1% of the simulations, the corresponding simulation was totally excluded from the analysis. When several equally parsimonious trees were found, only the first one was used. We did not do Bayesian trees because of the enormous computational time required for doing enough number of generations of all simulations performed.

For each alignment length, alignment strategy, and phylogenetic method, 300 simulations were run in a grid of 24 processors. The symmetric difference or Robinson-Foulds (Robinson and Foulds, 1981) topological distance from the calculated tree to the real tree was obtained using Vanilla 1.2 (Drummond and Strimmer, 2001), and the average of all simulations calculated. This program reports half the number of total discordant clades between two trees. For bootstrap analyses, 100 bootstraps were calculated. Due to heavy computational requirements of the bootstrap analyses, the number of simulations was reduced to 150. We checked that a higher number of bootstraps and simulations did not improve the accuracy of the bootstrap results. Bootstrap values were separately calculated for right and wrong partitions of the tree with the help of Bioperl functions (Stajich et al., 2002). Statistical differences among Robinson-Foulds distances in different alignment conditions were detected by the Tukey-Kramer test with an alpha level of 0.05 using the JMP package version 5.1 (SAS Institute, Cary, NC).

Results and Discussion

General Alignment Strategy: Complete versus Gblocks Alignments

The differences in alignments produced by different methods can be appreciated in Figure 2. A fragment of the alignment of simulated sequences (Fig. 2a) was stripped of gaps and realigned by ClustalW (Fig. 2b), Mafft (Fig. 2c), and Probcons (Fig. 2d). As it has been noted before (Higgins et al., 2005), ClustalW tends to produce more compact alignments. That is, ClustalW generates many divergent regions that are almost devoid of gaps, resulting in a relatively simple alignment (Higgins et al., 2005). This can be clearly appreciated in the most problematic region in the center of this alignment (Fig. 2b). Although Mafft also tends to make alignments more compact than the real ones (Fig. 2c), the deviation from the real situation is not as large as with ClustalW, at least with default gap penalties. Probcons produces the least compact alignments of the three programs tested (Fig. 2d). For example, simulations from asymmetric trees with divergence $\times 1$, which had an average original length of 1097 positions, were compacted to an average of 966 positions by Probcons, to 904 positions by Mafft and to 862 positions by ClustalW (Table 1). Similar relative degrees of compression were obtained in other types of simulations.

Gblocks removes problematic regions of a multiple alignment according to a number of rules. First, blocks selected for inclusion must be free from a large number of contiguous nonconserved positions, must be flanked by highly conserved positions, and must have a minimum length, as controlled by the corresponding parameters (see Materials and Methods). In addition, positions with gaps can be removed either always or only when more than half of the sequences contain gaps (Castresana, 2000). The latter parameter has a large influence on the total number of selected positions. We have used Gblocks in simulated realigned sequences with two different conditions. The condition that we call stringent does not allow any gap position. The relaxed condition allows gap positions if they are present in less than half of the sequences, and it is also less restrictive in the other parameters (see Materials and Methods). The effect of the two different parameter sets of Gblocks selection can be appreciated in Figure 2, for ClustalW (Fig. 2b), Mafft (Fig. 2c), and Probcons alignments (Fig. 2d). In both cases, the relaxed parameters (grey blocks) allow the selection of more positions than the stringent parameters (white blocks). Table 1 shows the average number of positions of the complete alignments and the percentage of positions left after treatment with Gblocks with the two different parameter sets. Values in this table are for the asymmetric tree, but similar values were found for other trees.

In order to infer which type of alignment algorithm (ClustalW, Mafft, or Probcons) and which treatment of the resulting alignment (no treatment or Gblocks treatment with stringent or relaxed conditions) was best for phylogenetic analysis, we calculated phylogenetic trees from all these alignments, and measured the topological distance with respect to the real tree. Figure 3 shows, for the simulations with the asymmetric tree, the average topological distances to the real tree from the trees generated with ClustalW alignments, with and without the use of Gblocks. In addition, the distance to the tree obtained from the Gblocks complementary alignment (that is, the alignment resulting after concatenation of all the blocks rejected by Gblocks) is also shown.

TABLE 1. Average number of positions of the complete alignments and the average percentage of positions selected by Gblocks with relaxed and stringent conditions. Simulation of sequences was done following the asymmetric tree and the heterogeneity pattern of the NAD2 protein concatenated two times.

	ClustalW			Mafft			Probcons		
Divergence	Total length	% Gblocks relaxed	% Gblocks stringent	Total length	% Gblocks relaxed	% Gblocks stringent	Total length	% Gblocks relaxed	% Gblocks stringent
$\times 0.5$	826.6	79.4	54.3	852.5	74.2	51.6	871.8	70.3	50.9
$\times 1$	862.4	64.2	42.0	903.7	59.0	39.8	966.4	51.8	37.6
$\times 2$	901.8	46.4	30.2	961.7	42.9	28.4	1117.9	34.7	24.5

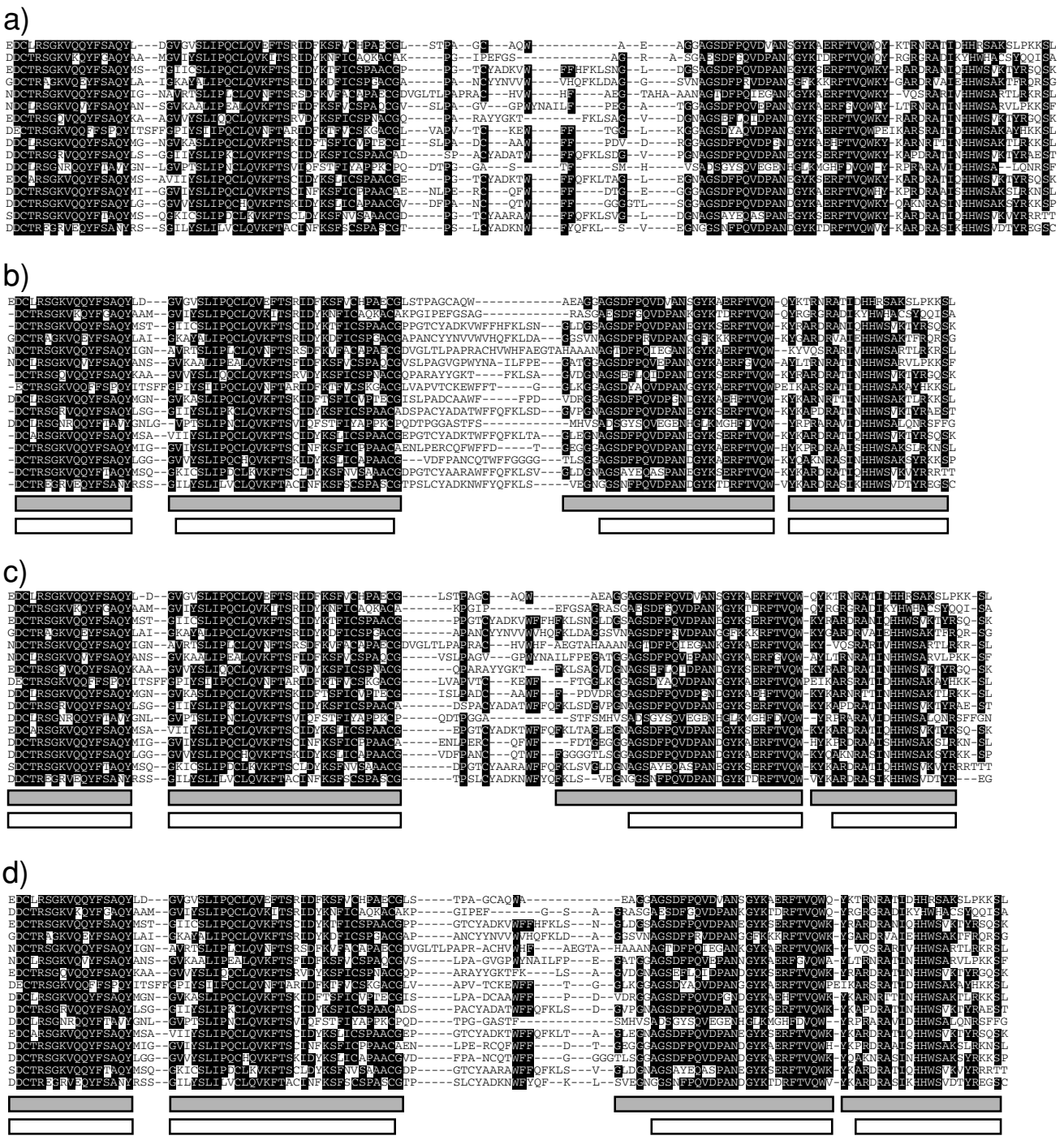

FIGURE 2. Fragment of a simulated alignment (a) and the realignment of the same sequences (after gap removal) by ClustalW (b), Mafft (c), and Probcons (d). The simulation corresponds to an asymmetric tree with divergence ×1. The blocks below each alignment represent the fragments selected by Gblocks with relaxed conditions (grey blocks) and with stringent conditions (white blocks). Positions of the alignments where more than 50% of the sequences are identical are shown with black boxes.

Figure 4 represents for each tree (and for two representative lengths, 800 and 3200 amino acids, as representatives of single-gene and concatenated-gene phylogenies) the best alignment strategies after statistically comparing the average topological distances by means of the Tukey-Kramer test. An overview of these two figures shows that, when the alignments are cleaned by Gblocks with any of the two parameter sets used (dotted lines in Figure 3), the topological distance to the real tree decreases with respect to the complete alignment (solid, red line) in almost all divergences and alignment lengths tested, and with the three tree reconstruction methods used:

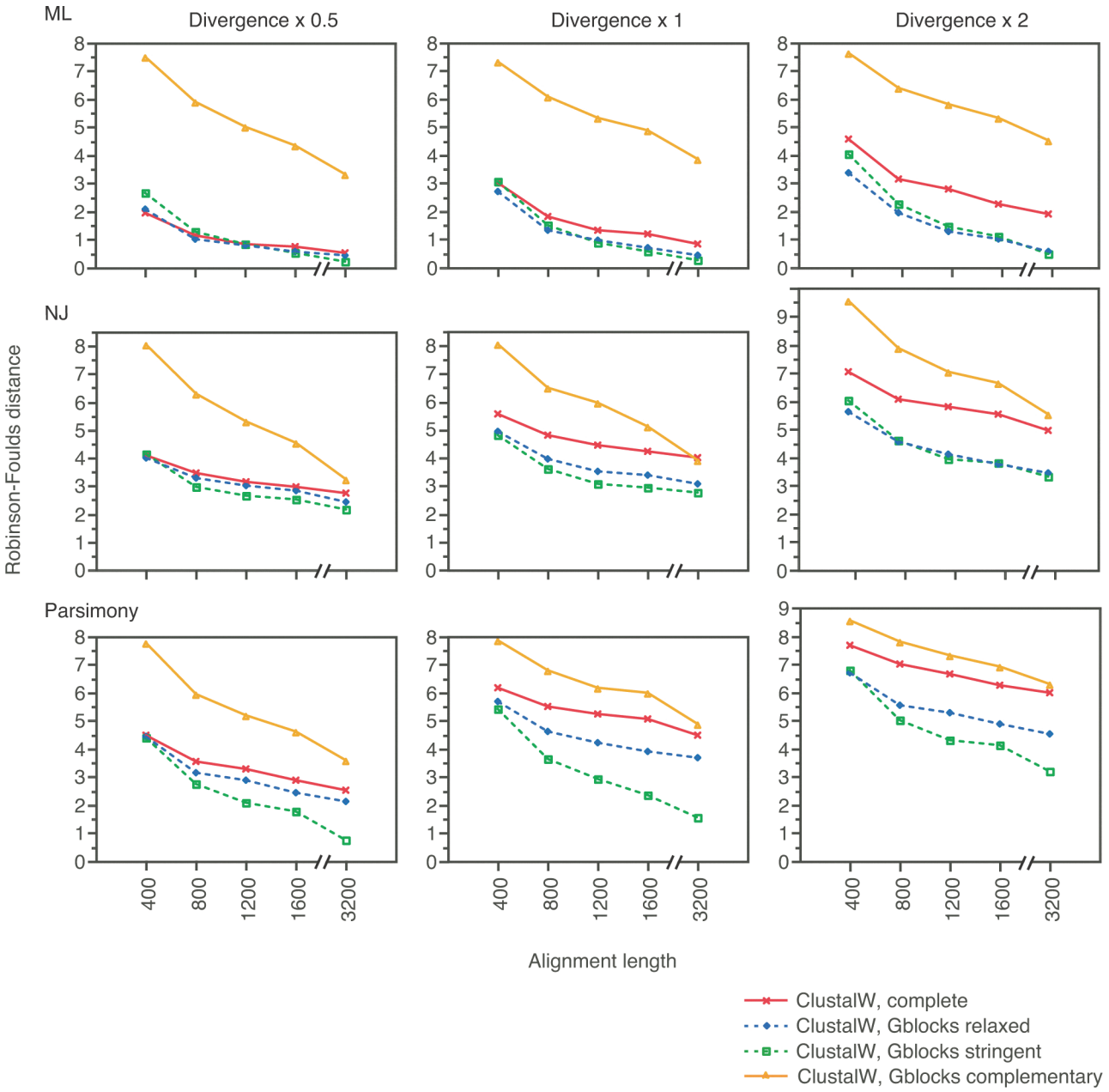

FIGURE 3. Average Robinson-Foulds distances to the real tree from the tree calculated with ClustalW complete alignments (solid, red line with crossed symbols), the same alignments after treatment with Gblocks relaxed (dotted, blue line with diamonds) and stringent (dotted, green line with squared symbols) conditions, and the complementary alignments of the Gblocks relaxed alignment (solid, orange line with triangles). The asymmetric tree with three different divergence levels was used for the simulations with different alignment lengths. Trees were reconstructed by ML, NJ, and parsimony.

ML, NJ, and parsimony. The improvement in topological accuracy upon Gblocks treatment is more noticeable for the highest divergences (×2). This is expected since there are more problematic blocks in these alignments, as shown by the lower percentage of positions selected by Gblocks (Table 1). In addition, the improvement from Gblocks treatment is particularly large for NJ and parsimony. These two methods produce quite poor topologies when using the complete alignments but, upon using Gblocks, particularly with the most stringent conditions (green line, squared symbols), there is a substantial gain in topological accuracy. ML produces the overall best trees (see also below) although, in the lowest divergence (×0.5), there is almost no difference in topological quality between the Gblocks and the complete alignments. In fact, for short genes (400 to 800 amino acids) the complete alignment gives rise to better trees than the Gblocks alignments, although there is no statistical difference between the complete alignment and the Gblocks alignment with relaxed parameters (Fig. 4).

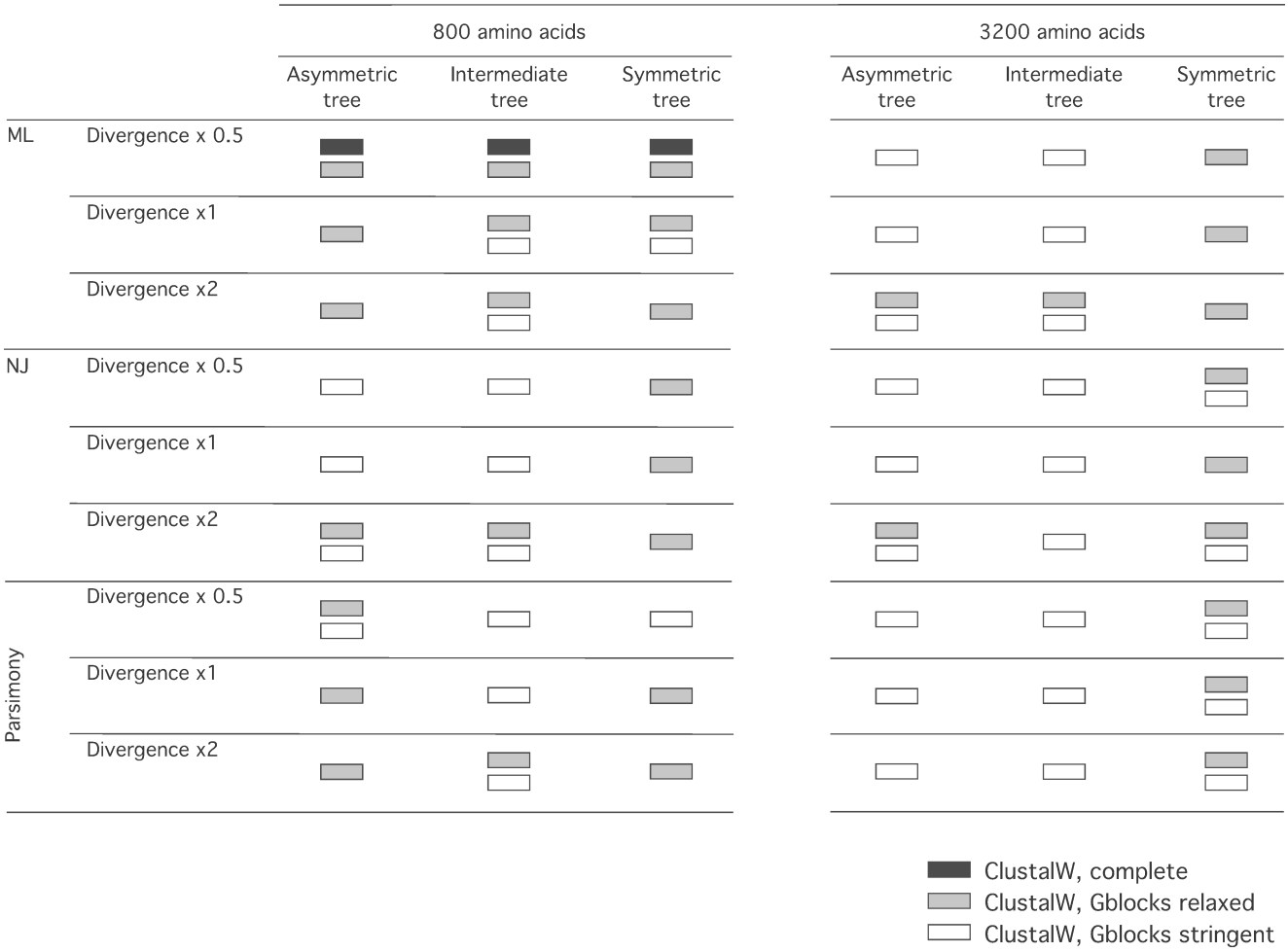

FIGURE 4. ClustalW alignment strategies that give rise to the statistically best topologies. When two or more strategies do not show statistical differences in Robinson-Foulds distances, all equivalent strategies are represented. The complete alignment is represented by a black block, and the relaxed and stringent Gblocks strategies by grey and white blocks, respectively.

It is thus shown from the example above that the removal of divergent and problematic regions of an alignment is, in principle, beneficial for phylogenetic analyses of relatively divergent sequences. In fact, it is true, as previously argued (Aagesen, 2004; Lee, 2001), that there is some phylogenetic information in the blocks removed by methods like Gblocks. This can be appreciated in Figure 3, which shows the topological distances to the real trees from the trees obtained with the blocks excluded by Gblocks (complementary alignment; solid, orange line). These distances, although very large, become quite reduced for long alignments, indicating that trees obtained from the complementary regions are not random; that is, there is some phylogenetic information in the regions rejected by Gblocks. However, what seems to matter is not the total phylogenetic signal but the signal-to-noise ratio. Despite the relatively simple simulations performed, regions excluded by Gblocks seem to add more noise than signal, thus lowering the quality of the trees from the complete alignments with respect to the Gblocks-cleaned alignments.

Similar conclusions about the beneficial effect of Gblocks can be drawn from Mafft alignments of the same asymmetric trees (Figs. 5 and 6). In this case, Gblocks is not an advantage over the complete alignment in the two most conserved alignments ($\times 0.5$ and $\times 1$) when using the ML method although, again, Gblocks relaxed and the complete alignments are not statistically different. The picture for Probcons (Fig. 1 of the online Appendix, available at http://systematicbiology.org) is similar to that for Mafft. Figure 2 of the online Appendix shows a comparison of the three alignment programs with default gap costs, using the trees produced after Gblocks cleaning with relaxed conditions. Under the conditions of these simulations, ClustalW is slightly worse, regarding the trees produced, than the two other programs. The performances of Mafft and Probcons are very similar, and only for NJ and parsimony Probcons alignments work slightly better. Probcons, however, is highly demanding in computational time. Thus, for the rest of the tests we only compared the performances of ClustalW and Mafft.

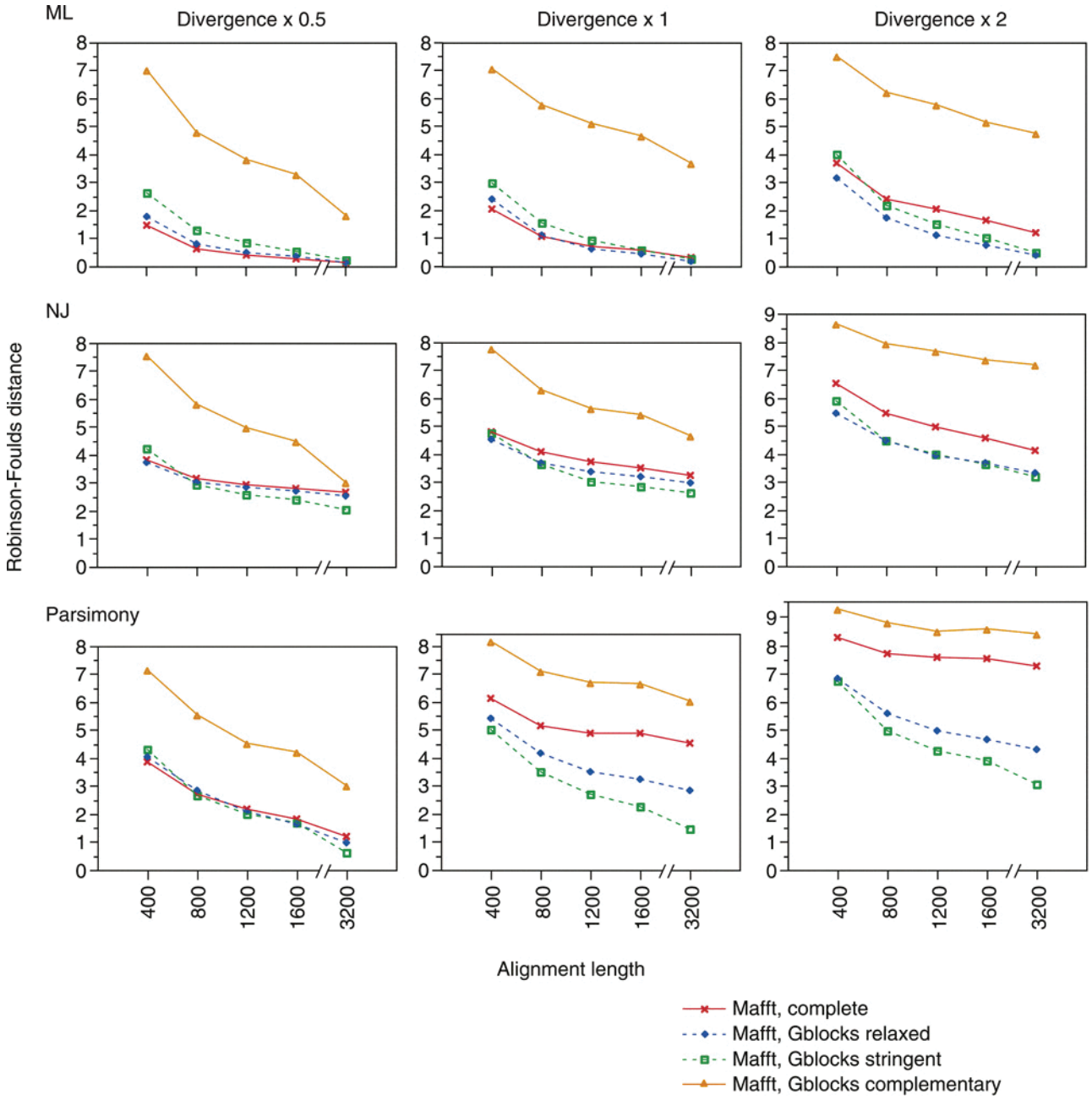

FIGURE 5. Average Robinson-Foulds distances to the real tree from the tree calculated with Mafft complete alignments (solid, red line with crossed symbols), the same alignments after treatment with Gblocks relaxed (dotted, blue line with diamonds) and stringent (dotted, green line with squared symbols) conditions, and the complementary alignments of the Gblocks relaxed alignment (solid, orange line with triangles). The asymmetric tree with three different divergence levels was used for the simulations with different alignment lengths. Trees were reconstructed by ML, NJ, and parsimony.

The results for the symmetric and intermediate trees of both alignment algorithms are shown in the corresponding columns of Figures 4 and 6 for the ClustalW and Mafft methods, respectively (and in Figures 3 to 6 in the online Appendix for all alignment lengths). Two results are noteworthy from these analyses. First, differences in phylogenetic performance between different alignments derived from symmetric trees are quantitatively smaller, in agreement with a previous work (Ogden and Rosenberg, 2006). See, for example, the similarity of the three graphs of ML trees of ClustalW alignments (Fig. 3 in the online Appendix). Second, in these trees there are two conditions where the Gblocks alignments produce ML trees that are statistically worse than the complete alignments: the symmetric and intermediate trees of divergence $\times 1$ with Mafft alignments of 800 amino acids (Fig. 6). These are the only two conditions where we observed this. However, we do not think that this justifies

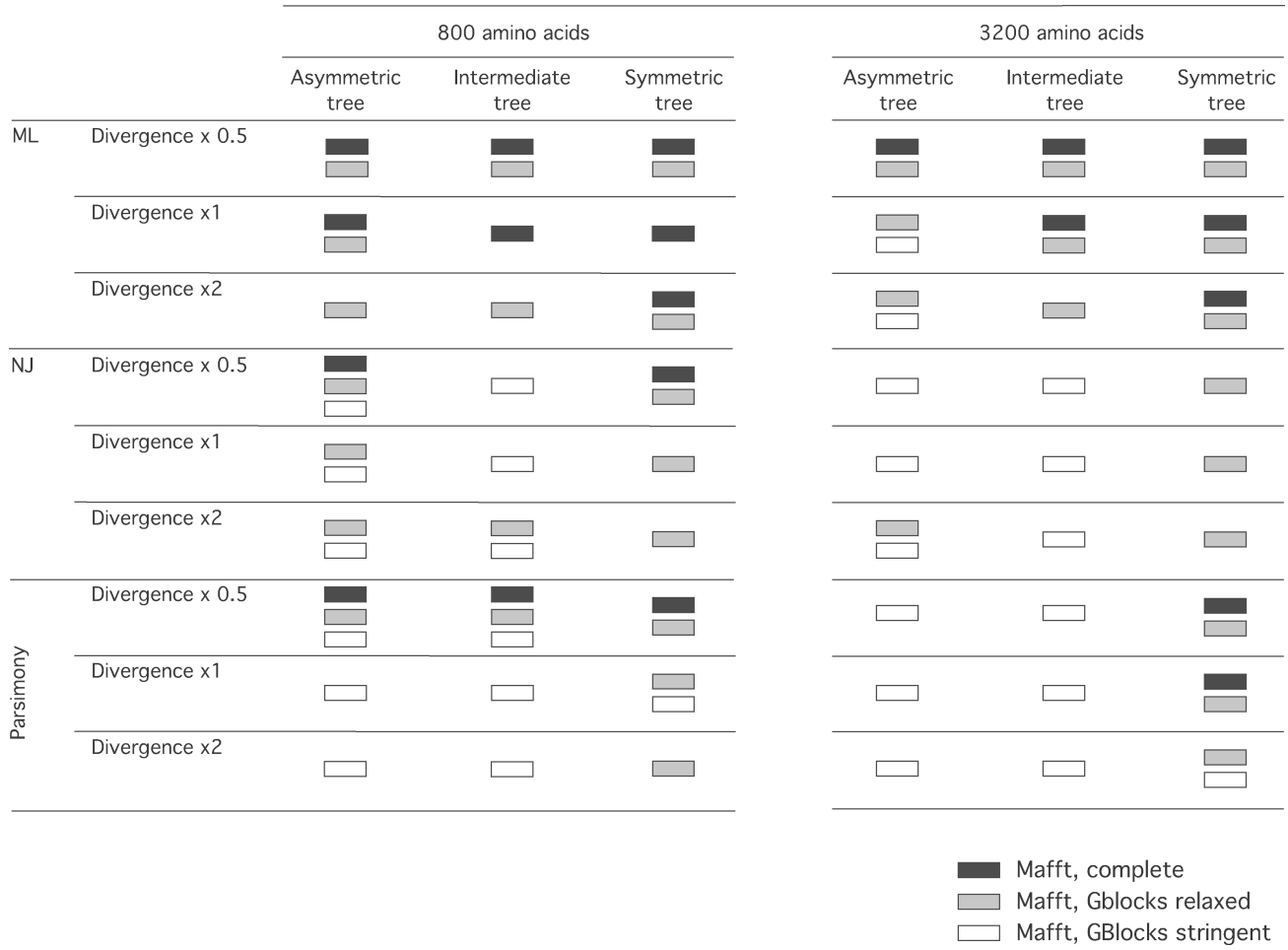

FIGURE 6. Mafft alignment strategies that give rise to the statistically best topologies. When two or more strategies do not show statistical differences in Robinson-Foulds distances, all equivalent strategies are represented. The complete alignment is represented by a black block, and the relaxed and stringent Gblocks strategies by grey and white blocks, respectively.

not using Gblocks in these types of trees, even if we could know the shape of the tree in advance. In real alignments, evolution must be much more complex than what we simulated. For example, we did not simulate biased amino acid compositions (Castresana et al., 1998a) or different models of evolution in different parts of trees (Philippe and Laurent, 1998), all of which will have stronger biasing effects in nonconserved blocks. Because the difference in topological accuracy between the Gblocks and the complete alignments is very small in these two conditions, it is very likely that the addition of any of these effects in the simulations would have made both the Gblocks relaxed and complete alignments of at least equal performance.

All simulations shown so far were performed following a pattern of rate variation of the NAD2 protein. To test the influence of different rate patterns, we used in the simulations profiles derived from two other proteins (NAD4 and COG0285). From the Mafft alignments of these simulations we calculated the corresponding ML trees (Fig. 7 in the online Appendix). Different patterns (and thus different percentages of block selection) gave rise to different performances of the complete and the Gblocks alignments, but the results were similar in relative terms. We also tested the performance of a different gap model, in which gaps were introduced homogeneously along the alignment, instead of using two different gap thresholds in different regions of the alignments (see Materials and Methods). The results were again similar with the simpler gap strategy, as shown for the ML reconstruction of the asymmetric trees (Fig. 8 of the Online appendix).

Phylogenetic Methods Used

The data shown above indicate that ML is the phylogenetic method that best extracts reliable information from problematic alignment regions, since trees derived from complete alignments are relatively good. This contrasts with the trees obtained by NJ and parsimony, which are quite poor from the complete alignments, indicating that they greatly benefited from the use of Gblocks. ML is also the method that produces the overall best trees, in agreement with previous simulation analysis (see references

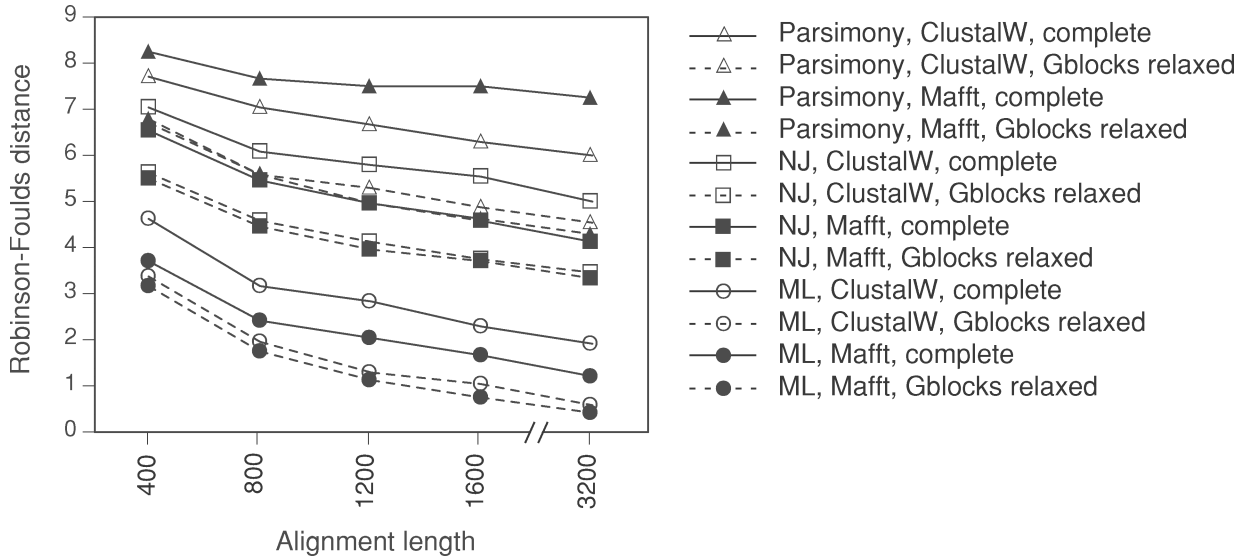

FIGURE 7. Average Robinson-Foulds distances to the real tree from the tree calculated with Mafft complete (solid line, solid symbols) and ClustalW complete alignments (solid line, empty symbols). The tree distances obtained with the same alignments after treatment with Gblocks with relaxed conditions (dotted lines) are also shown. Trees were reconstructed by ML (circles), NJ (squares), and parsimony (triangles). The most divergent asymmetric tree was used for the simulations.

in Felsenstein, 2004). To show this, Figure 7 presents the superimposed graphs for the most divergent asymmetric tree as an example. The better performance of ML in all alignment conditions is clearly appreciated in this graph.

Short versus Long Alignments

Alignment length turned out to be a very important factor to be taken into account when deciding the best alignment cleaning strategy. Figures 3 and 5 show that, in general, for shorter alignments the best Gblocks condition is the relaxed one, whereas for longer alignments the stringent condition tends to work better. This can also be appreciated by comparing the slopes of the graphs corresponding to the complete alignments, and those of the Gblocks alignments with relaxed and stringent conditions. The slope downwards (towards better trees) is less pronounced for the complete alignments and more pronounced for Gblocks with stringent conditions. This means that for single genes (400 to 800 amino acids) the gain in signal-to-noise ratio after elimination of problematic blocks may not compensate the total loss of information. However, for longer alignments, for example, those used in phylogenomic studies where several genes are concatenated (Delsuc et al., 2005; Jeffroy et al., 2006), there is enough total information so that selecting the best pieces with Gblocks using the stringent conditions allows to get closer to the real tree. This basic tendency is observed under all simulation conditions we tested.

Bootstrap Support in Trees Obtained from Gblocks Alignments

Previous performance tests of Gblocks with real data showed that Gblocks alignments obtained less support in ML analysis, because the number of trees not significantly different from the ML tree was smaller in the complete alignment than in the Gblocks alignment (Castresana, 2000). Later, in numerous studies in our group and in other groups, the same effect was observed using bootstrap values of NJ trees, which were lower in the Gblocks alignments. Our simulations reproduced the same behavior again. In NJ trees obtained from 100 bootstrap samples, the average bootstrap support of all partitions was higher for the complete alignments, and lower for Gblocks alignments (Fig. 8). However, the same simulations (see topological distances of NJ trees in Figures 3 and 5) showed that the best trees were obtained with Gblocks conditions and the worse topologies with the complete alignments, thus following the opposite direction, regarding quality, to the bootstrap values, at least for the maximum divergence. A similar trend was found for NJ trees of simulations with symmetric trees (Fig. 9 of the online Appendix) and for bootstrapped ML trees (Fig. 10 of the online Appendix). One may think that the bootstraps of Gblocks trees are lower due to the smaller length of the Gblocks alignments, but it is still very paradoxical that the best topology is associated to a lower bootstrap.

The explanation for this contradictory behavior of Gblocks may be that divergent and problematic alignment regions are biased towards an erroneous topology (Lake, 1991). This could happen if the initial guide tree used in the progressive alignment methods is conducting very strongly the alignment in the divergent and most gappy regions, where alignment programs may easily create similarity at the expense of homology (Higgins et al., 2005). In addition, when alignment software is faced with an ambiguous alignment decision, the algorithmic solution makes consistent but arbitrary decisions

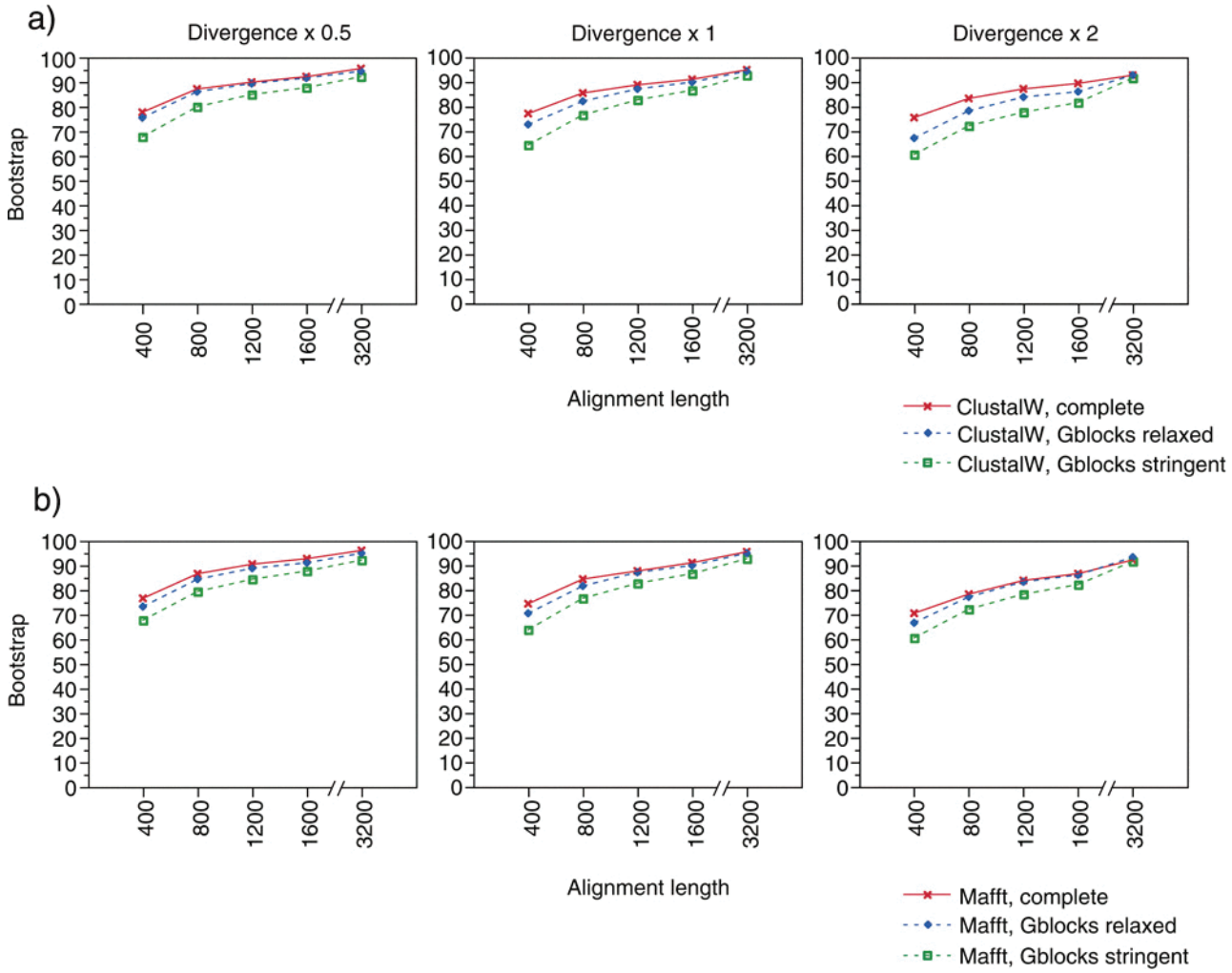

FIGURE 8. Average bootstrap values of NJ trees obtained from ClustalW (a) and Mafft (b) alignments simulated from the asymmetric tree with three different divergence levels. Complete (solid, red line), Gblocks relaxed (dotted, blue line with diamonds), and Gblocks stringent (dotted, green line with squared symbols) alignments are shown.

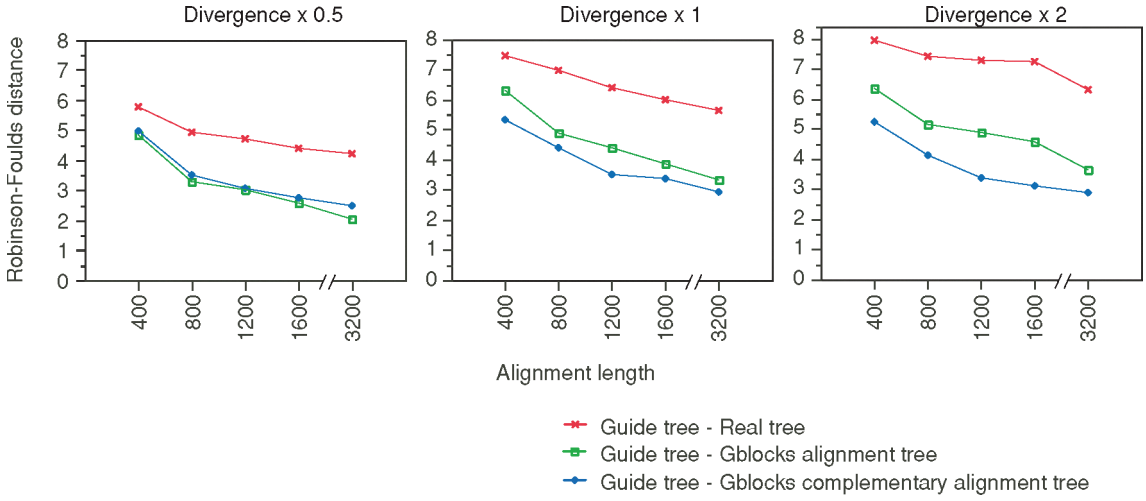

FIGURE 9. Average Robinson-Foulds distances from the ClustalW guide tree to the real tree (red line with crossed symbols), from the guide tree to the NJ tree of the Gblocks alignment with relaxed conditions (green line with squared symbols), and from the guide tree to the NJ tree of the complementary positions of the same Gblocks alignment (blue line with diamonds). The asymmetric tree with three different divergence levels was used for the simulations.

that bias the support indices. That is, this repeated alignment decisions will increase the bootstrap support, and this bias will be stronger in the most divergent regions, where there is more uncertainty. Three results are consistent with this possibility. Firstly, we have observed in our simulations that the initial guide dendrogram used by ClustalW is indeed very different from the real tree, as measured by the Robinson-Foulds distance of both trees (Fig. 9). If all divergent regions tend to easily reproduce this initial dendrogram, we would expect that the guide tree is more similar to the tree obtained from the Gblocks excluded regions than to the Gblocks alignment. Figure 9 shows that this is the case, particularly in the most divergent simulations. Secondly, we see that the effect of increased bootstrap support in the complete alignment with respect to the Gblocks alignments is higher in ClustalW, which highly depends on the initial dendrogram, than in Mafft (Fig. 8). For example, in simulations of 400 amino acids and at $\times 2$ divergence, there is an increase from 60% to 76% bootstrap support in ClustalW when comparing the Gblocks stringent and complete alignments, and only from 60% to 70% in Mafft. In the latter method, the successive iterations of the alignment algorithm may make the final alignment more independent from the initial crude dendrogram, thus explaining that trees generated from these alignments are slightly less biased. And thirdly, when we calculated separately bootstraps of right and wrong partitions for each tree we observe, apart from lower values for wrong partitions, a slightly higher bias in them (Fig. 11 of the online Appendix). The bias is also present in the right partitions, probably because some of the recurrent software decisions in the divergent regions are actually correct. Thus, the bias coming from divergent regions seems to increase the bootstrap of all partitions, although the effect is slightly larger in the wrong ones. All this indicates that bootstrap support cannot be used as a measure of reliability of the tree topology when divergent regions are present in the alignment.

CONCLUSIONS

We have shown, under the conditions of these simulations, that the information contained in divergent and ambiguously aligned regions of multiple alignments is, in general, not beneficial for phylogenetic reconstruction. Thus, using Gblocks or a similar method for removing problematic blocks seems to be justified for phylogenetic analysis, particularly for divergent alignments. In this work, we have used simulations of moderately divergent and very heterogeneous proteins, which are typically used in deep phylogenies (i.e., bacterial groups, eukaryotes lineages, metazoan phyla). However, we do not know how removal of blocks would affect more conserved and less heterogeneous alignments. We have also not tested how a finer tuning of parameters of alignment programs and Gblocks may improve the phylogenies. Although we have only used protein alignments, the same conclusions are expected to apply to protein-coding DNA alignments of similar divergence. On the other hand, although we predict that the general conclusion that ambiguously aligned regions in any data set are best excluded when they provide more noise than signal, rRNA alignments as well as alignments from noncoding DNA have very different features from coding alignments, and our simulations were not specifically designed to explore the properties of these kinds of sequences. However, our purpose in this work is not giving strict rules about the best alignment strategy and associated parameters. Rather, our simulations are mainly informative about general tendencies. Thus, in the following we summarize important tendencies observed in our simulations and give some general rules regarding the best alignment strategy that can be applied to real situations of protein alignments.

NJ and parsimony seem to be unable to extract useful phylogenetic information from the problematic alignment regions, because the complete alignments are always much worse than the Gblocks treated alignments, so using Gblocks seems particularly advisable for these methods. Most probably, these two methods are not able to take into account the multiple substitutions that occur in these excessively saturated blocks. On the other hand, ML, less affected by saturation, is able to extract some information from these blocks, since in some conditions the complete alignments are similar or even better than the Gblocks alignments. However, the misidentified homology that may occur in these regions affects all phylogenetic methods, which may explain why using Gblocks is more beneficial at high divergences for all methods.

Regarding the use of stringent or relaxed conditions for Gblocks, two important rules can be extracted from our analysis. First, for ML trees relaxed conditions of Gblocks seem to give rise to better trees, whereas for NJ and parsimony stringent conditions are better. Second, alignment length is a crucial parameter to be taken into account. For short alignments, such as in studies of single short genes, the removal of blocks by Gblocks may leave too few positions, so in these cases it may be better to use very relaxed conditions of Gblocks. In the shortest alignments, which have very little information, use of Gblocks may be even detrimental. At any rate, one should be aware that with this type of short alignments it is only possible to obtain a very approximate topology, possibly quite distant from the real tree. For phylogenomic studies, where there is enough information from the concatenation of several genes (Jeffroy et al., 2006), the use of Gblocks with stringent conditions tends to give rise to the best phylogenetic trees.

ACKNOWLEDGMENTS

This work was supported financially by a research grant in bioinformatics from the Fundación BBVA (Spain), and grant number BIO2002-04426-C02-02 from the Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica (I+D+I) of the MEC, cofinanced with FEDER funds. We thank V. Soria-Carrasco for useful technical assistance, and three anonymous reviewers, K. Kjer, and R.D.M. Page for critical comments that helped improve the manuscript.

REFERENCES

Aagesen, L. 2004. The information content of an ambiguously alignable region, a case study of the trnL intron from the Rhamnaceae. Organ. Divers. Evol. 4:35–49.

Blackshields, G., I. M. Wallace, M. Larkin, and D. G. Higgins. 2006. Analysis and comparison of benchmarks for multiple sequence alignment. In Silico Biol. 6:321–339.

Castresana, J. 2000. Selection of conserved blocks from multiple alignments for their use in phylogenetic analysis. Mol. Biol. Evol. 17:540–552.

Castresana, J., G. Feldmaier-Fuchs, and S. Pääbo. 1998a. Codon reassignment and amino acid composition in hemichordate mitochondria. Proc. Natl. Acad. Sci. USA 95:3703–3707.

Castresana, J., G. Feldmaier-Fuchs, S. Yokobori, N. Satoh, and S. Pääbo. 1998b. The mitochondrial genome of the hemichordate *Balanoglossus carnosus* and the evolution of deuterostome mitochondria. Genetics 150:1115–1123.

Dayhoff, M. O., R. M. Schwartz, and B. C. Orcutt. 1978. A model of evolutionary change in proteins. Pages 345–352 *in* Atlas of protein sequence structure (M. O. Dayhoff, ed.) National Biomedical Research Foundation, Washington, D.C.

Delsuc, F., H. Brinkmann, and H. Philippe. 2005. Phylogenomics and the reconstruction of the tree of life. Nat. Rev. Genet. 6:361–375.

Do, C. B., M. S. Mahabhashyam, M. Brudno, and S. Batzoglou. 2005. ProbCons: Probabilistic consistency-based multiple sequence alignment. Genome Res. 15:330–340.

Drummond, A., and K. Strimmer. 2001. PAL: An object-oriented programming library for molecular evolution and phylogenetics. Bioinformatics 17:662–663.

Edgar, R. C. 2004. MUSCLE: Multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res. 32:1792–1797.

Felsenstein, J. 1989. PHYLIP—Phylogeny inference package (version 3.4). Cladistics 5:164–166.

Felsenstein, J. 2004. Inferring phylogenies. Sinauer Associates, Sunderland, Massachusetts.

Feng, D. F., and R. F. Doolittle. 1987. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. J. Mol. Evol. 25:351–360.

Fleissner, R., D. Metzler, and A. von Haeseler. 2005. Simultaneous statistical multiple alignment and phylogeny reconstruction. Syst. Biol. 54:548–561.

Gatesy, J., R. DeSalle, and W. Wheeler. 1993. Alignment-ambiguous nucleotide sites and the exclusion of systematic data. Mol. Phylogenet. Evol. 2:152–157.

Geiger, D. L. 2002. Stretch coding and block coding: Two new strategies to represent questionably aligned DNA sequences. J. Mol. Evol. 54:191–199.

Grundy, W. N., and G. J. Naylor. 1999. Phylogenetic inference from conserved sites alignments. J. Exp. Zool. 285:128–139.

Guindon, S., and O. Gascuel. 2003. A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. Syst. Biol. 52:696–704.

Gutell, R. R., N. Larsen, and C. R. Woese. 1994. Lessons from an evolving rRNA: 16S and 23S rRNA structures from a comparative perspective. Microbiol. Rev. 58:10–26.

Henikoff, S., and J. G. Henikoff. 1994. Protein family classification based on searching a database of blocks. Genomics 19:97–107.

Herrmann, G., A. Schon, R. Brack-Werner, and T. Werner. 1996. CONRAD: A method for identification of variable and conserved regions within proteins by scale-space filtering. Comput. Appl. Biosci. 12:197–203.

Higgins, D. G., G. Blackshields, and I. M. Wallace. 2005. Mind the gaps: Progress in progressive alignment. Proc. Natl. Acad. Sci. USA 102:10411–10412.

Jeffroy, O., H. Brinkmann, F. Delsuc, and H. Philippe. 2006. Phylogenomics: The beginning of incongruence? Trends Genet. 22:225–231.

Jones, D. T., W. R. Taylor, and J. M. Thornton. 1992. The rapid generation of mutation data matrices from protein sequences. Comput. Appl. Biosci. 8:275–282.

Katoh, K., K. Kuma, H. Toh, and T. Miyata. 2005. MAFFT version 5: Improvement in accuracy of multiple sequence alignment. Nucleic Acids Res. 33:511–518.

Katoh, K., K. Misawa, K. Kuma, and T. Miyata. 2002. MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform. Nucleic Acids Res. 30:3059–3066.

Kjer, K. M. 1995. Use of rRNA secondary structure in phylogenetic studies to identify homologous positions: an example of alignment and data presentation from the frogs. Mol. Phylogenet. Evol. 4:314-330.

Lake, J. A. 1991. The order of sequence alignment can bias the selection of tree topology. Mol. Biol. Evol. 8:378–385.

Lassmann, T., and E. L. Sonnhammer. 2005. Kalign—An accurate and fast multiple sequence alignment algorithm. BMC Bioinformatics 6:298.

Lee, M. S. 2001. Unalignable sequences and molecular evolution. Trends Ecol. Evol. 16:681–685.

Löytynoja, A., and M. C. Milinkovitch. 2001. SOAP, cleaning multiple alignments from unstable blocks. Bioinformatics 17:573–574.

Lunter, G., I. Miklos, A. Drummond, J. L. Jensen, and J. Hein. 2005. Bayesian coestimation of phylogeny and sequence alignment. BMC Bioinformatics 6:83.

Lutzoni, F., P. Wagner, V. Reeb, and S. Zoller. 2000. Integrating ambiguously aligned regions of DNA sequences in phylogenetic analyses without violating positional homology. Syst. Biol. 49:628–651.

Morrison, D. A., and J. T. Ellis. 1997. Effects of nucleotide sequence alignment on phylogeny estimation: A case study of 18S rDNAs of apicomplexa. Mol. Biol. Evol. 14:428–441.

Needleman, S. B., and C. D. Wunsch. 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., D. G. Higgins, and J. Heringa. 2000. T-Coffee: A novel method for fast and accurate multiple sequence alignment. J. Mol. Biol. 302:205–217.

Nuin, P. A., Z. Wang, and E. R. Tillier. 2006. The accuracy of several multiple sequence alignment programs for proteins. BMC Bioinformatics 7:471.

Ogden, T. H., and M. S. Rosenberg. 2006. Multiple sequence alignment accuracy and phylogenetic inference. Syst. Biol. 55:314–328.

Pesole, G., M. Attimonelli, G. Preparata, and C. Saccone. 1992. A statistical method for detecting regions with different evolutionary dynamics in multialigned sequences. Mol. Phylogenet. Evol. 1:91–96.

Philippe, H., and J. Laurent. 1998. How good are deep phylogenetic trees? Curr. Opin. Genet. Dev. 8:616–623.

Redelings, B. D., and M. A. Suchard. 2005. Joint Bayesian estimation of alignment and phylogeny. Syst. Biol. 54:401–418.

Robinson, D. F., and L. R. Foulds. 1981. Comparison of phylogenetic trees. Math. Biosci. 53:131–147.

Rodrigo, A. G., P. R. Bergquist, and P. L. Bergquist. 1994. Inadequate support for an evolutionary link between the Metazoa and the Fungi. Syst. Biol. 43:578–584.

Smythe, A. B., M. J. Sanderson, and S. A. Nadler. 2006. Nematode small subunit phylogeny correlates with alignment parameters. Syst. Biol. 55:972–992.

Stajich, J. E., et al. 2002. The Bioperl toolkit: Perl modules for the life sciences. Genome Res. 12:1611–1618.

Stoye, J., D. Evers, and F. Meyer. 1998. Rose: Generating sequence families. Bioinformatics 14:157–163.

Strimmer, K., and A. von Haeseler. 1996. Quartet puzzling: A quartet maximum-likelihood method for reconstructing tree topologies. Mol. Biol. Evol. 13:964–969.

Swofford, D. L., G. J. Olsen, P. J. Waddell, and D. M. Hillis. 1996. Phylogenetic inference. Pages 407–514 *in* Molecular systematics (D. M. Hillis, C. Moritz, and B. K. Mable, eds.). Sinauer Associates, Sunderland, Massachusetts.

Tatusov, R. L., N. D. Fedorova, J. D. Jackson, A. R. Jacobs, B. Kiryutin, E. V. Koonin, D. M. Krylov, R. Mazumder, S. L. Mekhedov, A. N. Nikolskaya, B. S. Rao, S. Smirnov, A. V. Sverdlov, S. Vasudevan, Y. I. Wolf, J. J. Yin, and D. A. Natale. 2003. The COG database: an updated version includes eukaryotes. BMC Bioinformatics 4:41.

Thompson, J. D., D. G. Higgins, and T. J. Gibson. 1994. 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.

Thompson, J. D., P. Koehl, R. Ripp, and O. Poch. 2005. BAliBASE 3.0: Latest developments of the multiple sequence alignment benchmark. Proteins 61:127–136.

Wheeler, W. 2001. Homology and the optimization of DNA sequence data. Cladistics 17:S3–S11.

Xia, X., Z. Xie, and K. M. Kjer. 2003. 18S ribosomal RNA and tetrapod phylogeny. Syst. Biol. 52:283–295.

Yang, Z. 1998. On the best evolutionary rate for phylogenetic analysis. Syst. Biol. 47:125–133.

Young, N. D., and J. Healy. 2003. GapCoder automates the use of indel characters in phylogenetic analysis. BMC Bioinformatics 4:6.

First submitted 7 February 2007; reviews returned 6 March 2007; final acceptance 24 March 2007

Associate Editor: Karl Kjer

Editors: Rod Page and Jack Sullivan