MBE Advance Access published April 1, 2014

TCS: A new multiple sequence alignment reliability measure to estimate alignment accuracy and improve phylogenetic tree reconstruction

Jia-Ming Chang[1,2,3], Paolo Di Tommaso[1,2], Cedric Notredame[1,2,*]

[1]Comparative Bioinformatics, Centre for Genomic Regulation (CRG), Dr. Aiguader 88, 08003 Barcelona, Spain

[2]Universitat Pompeu Fabra (UPF), Barcelona, Spain

[3]Current address: Institute of Human Genetics (IGH), UPR 1142, CNRS, 141 rue de la Cardonille, 34396 Montpellier, France

*To whom correspondence should be addressed. E-mail: Cedric.Notredame@crg.eu

ABSTRACT

Multiple sequence alignment (MSA) is a key modeling procedure when analyzing biological sequences. Homology and evolutionary modeling are the most common applications of MSAs. Both are known to be sensitive to the underlying MSA accuracy. In this work we show how this problem can be partly overcome using the transitive consistency score (TCS), an extended version of the T-Coffee scoring scheme. Using this local evaluation function we show that one can identify the most reliable portions of an MSA, as judged from BAliBASE and PREFAB structure based reference alignments. We also show how this measure can be used to improve phylogenetic tree reconstruction using both an established simulated dataset and a novel empirical yeast dataset. For this purpose, we describe a novel lossless alternative to site filtering that involves over-weighting the trustworthy columns. Our approach relies on the T-Coffee framework; it uses libraries of pairwise alignments to evaluate any third party MSA. Pairwise projections can be produced using fast or slow methods, thus allowing a trade-off between speed and accuracy. We compared TCS to HoT, GUIDANCE, Gblocks and trimAl and found it to lead to significantly better estimate of structural accuracy as well as more accurate phylogenetic trees.

Availability: TCS is part of the T-Coffee package, a freeware open source code can be downloaded from http://www.tcoffee.org/Packages/Stable/Latest and a web server is also available from http://tcoffee.crg.cat/tcs.

Introduction

Multiple sequence alignment (MSA) is an important initial step when deriving the most common biological models, including phylogenetic reconstruction, structural homology modeling and functional inference through domain profile comparisons. More than 100 publications describing novel MSA methods have been published over the last 30 years (Kemena and Notredame 2009). The accuracy of MSA is limited both by the problem complexity (known to be NP-Complete (Wang and Jiang 1994) in most useful formulations) and the difficulty of describing sequence homology in mathematical terms. These two hurdles have contributed to making MSA research an exceptionally active investigation field, with new fronts opening on a regular basis. One of the latest developments has been a gradual shift from systematic attempts in designing more accurate aligners towards the definition of accuracy indexes allowing objective identification of MSAs most trustworthy positions. The rationale of this approach is to privilege high quality data subsets over the maintenance of uncertain regions embedded within the full dataset.

The main reason why MSA reliability fluctuates lies in our limited capacity to meaningfully describe sequence homology, especially when dealing with remote homologues less than 20% identical (Sander and Schneider 1991; Rost 1999). At this level, homology signal tends to be saturated and lower than background noise. Aligners, however, are usually meant to maximize similarity and therefore tend to over-estimate identity (Notredame et al. 1998). This global phenomenon can be heterogeneous due to evolutionary rate differences across sites. Considering that MSAs are usually built around a unique homology model (typically a substitution matrix and an affine gap penalty scheme), it is reasonable to expect significant local variation in accuracy, especially between slow and fast evolving protein portions (typically the core and the loops). This problem is worsened by the reliance of most MSA packages onto dynamic programming (DP) pairwise comparison algorithms like Needlman and Wunsch (NW)

(Needleman and Wunsch 1970). NW estimates the optimal weighted edit score of two sequences and it delivers one unique pairwise alignment having this optimal score. In most implementations, whenever more than one optimal alignments exist, ties are broken arbitrarily and deterministically so that the algorithm always returns the same alignment. For instance, the code is often written in such a way that when at a given cell of the DP matrix, the best cost is the same when considering a gap in the top sequence or a gap in the bottom, the code arbitrarily selects the gap in the top sequence. This process is informally referred to as “low-road/high-road” resolution. The order in which ties are resolved therefore depends on the input sequence order. By swapping them, one systematically inverts the gap tiebreaks, which may significantly alter the pairwise output without changing the score. One can also obtain the same effect by flipping the sequences from left to right, a procedure used in the Heads-or-Tails (HoT) method (Landan and Graur 2007, 2008).

Given two moderately related sequences, these arbitrary tiebreaks have little consequence and usually only affect gaps edges. By contrast, they can have a dramatic effect on MSAs computed using a guide-tree based progressive approach, like most aligners do. With each tiebreak being the equivalent of a coin toss, it may happen that sub-alignments become incompatible as a consequence of un-concerted tiebreak decisions. This effect typically results in ragged indel columns, where a gap is extended to the left in half of the sequences and to the right in the other half, it may also suggest spurious duplications or motif expansions on large MSAs. The only solution would be to coordinate all tiebreaks, but doing so has a prohibitive CPU cost (Lipman et al. 1989). Another alternative, implemented in the PRANK algorithm, is to turn MSA computation into a sampling across the tiebreak space (Loytynoja and Goldman 2008; Blackburne and Whelan 2013). In PRANK all ties are broken randomly and one may gain some insight on the model robustness with respect to tiebreaks by re-computing the same MSA. Consistency based progressive evaluation is a simple and powerful alternative to this costly sampling strategy. In a progressive consistency framework (Notredame et al. 2000; Do et al. 2005; Roshan and Livesay 2006; Liu et al. 2010), MSAs are estimated by maximizing compatibility with a set of pre-computed pairwise alignments (library). This algorithm was first described along with the original T-Coffee package, in which the library was computed by feeding two alternative pairwise alignments for each pair of sequences (i.e. the two possible orders). By doing so, the algorithm was populating a pairwise library, which was then used to estimate the propensity of every pair of residue to be aligned, given its compatibility with the rest of the library.

Library based scoring schemes help coordinate tiebreaks, since they are better informed than substitution matrix scoring schemes and therefore more likely to remain compatible across the guide-tree. In a later implementation of this algorithm (Do et al. 2005), ProbCons authors went a bit further and used a pair-HMM to populate their library, selecting all pairs of residues having an alignment posterior probability higher than some empirical threshold. This procedure allows considering simultaneously both alternative and sub-optimal alignments. As noted by T-Coffee and ProbCons authors (Notredame et al. 2000; Do et al. 2005), the resulting consistency score for aligning two symbols reflects the support of the whole sequence dataset, and may therefore be used as a reliability indicator. More recently, HoT (Landan and Graur 2007) took advantage of the NW algorithm tiebreak sensitivity by comparing the direct MSA and its flipped version obtained through the same aligner. The authors did a structure-based validation and also showed that flipping sensitivity could be related to phylogenetic topological instability.

The main motivation of methods like HoT is to use instability estimations in order to identify the most reliable portions inside an MSA. This concept was recently taken a bit further in GUIDANCE (Penn et al. 2010), that uses alternative guide-trees obtained from bootstrap replicates to estimate alternative MSAs and turn their consistency into a reliability index. GUIDANCE was benchmarked on structure-based datasets and shown to outperform HoT. As an alternative to random guide trees, the authors of PARS (Kim and Ma 2011) have recently estimated DNA sequences MSA accuracy by comparing alternative projections of the same sequences in an MSA when removing in turn every sequence and realigning the remaining set. All these methods share a reliance on some form of data perturbation, meant to reveal MSA resilience, a bit like bootstrap does in phylogeny. Perturbations always require some form of CPU expensive resampling. By contrast, consistency based methods do not require re-sampling and simply rely on data self-consistency, established in a closed form computation. Consistency-based filtering was first described as a T-Coffee application named CORE (Consistency of Overall Residue Evaluation) (Notredame and Abergel 2003) later used as a mean to automatically filter out unreliable positions when doing homology modeling (Claude et al. 2004). Perturbation and consistency-based approaches differ significantly from the most common filtering procedures used to 'clean' unreliable positions. These other methods, like Gblocks (Castresana 2000; Talavera and Castresana 2007) or trimAl (Capella-Gutierrez et al. 2009), eliminate MSA positions on the basis of their conservation and indel propensities. They have been extensively validated on simulated datasets and are now routinely used as an attempt to improve phylogenetic reconstruction.

All these estimators share the same purpose: estimating unreliable positions in order to improve downstream modeling. To the best of our knowledge, no previous attempt of simultaneously evaluating the phylogenetic and structural potential of reliability index methods has yet been reported. On the one hand, Gblocks and trimAl were extensively benchmarked on phylogenetic simulated datasets. GUIDANCE, HoT and CORE, on the other hand, were only validated for their capacity to recognize structurally correct alignments. The closest to a phylogenetic validation was done in HoT (Landan and Graur 2007) where the authors did show that MSA instability (HoT wise) was correlated to phylogenetic instability. Addressing these two issues at once has become critical. Indeed, the recent development of phylogeny aware aligners like PRANK (Loytynoja and Goldman 2008) or SATe (Liu et al. 2009), is raising some issues on the suitability of structurally correct MSAs for phylogenetic trees reconstruction and vice-versa. This problem is especially obvious when considering the PRANK package that performs poorly on most structure based datasets even though other benchmarks suggest its superiority when doing phylogenetic reconstruction.

Our analysis tends to rule out this discrepancy and on most tested aligners, with the notable exception of PRANK, we did find some correlation between structural and phylogenetic modeling accuracy. We show here that our index, named TCS, is equally suitable for the identification of correctly aligned residues, as judged by structural analysis, and for the improvement of phylogenetic reconstruction through the selection of columns weighted by the TCS procedure. We find this effect to be similar across the two most widely used structural benchmark references (BAliBASE 3 (Thompson et al. 2005a) or PREFAB 4 (Edgar 2004b)), as well as when reconstructing phylogenies on simulated or empirical datasets. On our data, SATe, the best performing method is equally good at structural and phylogenetic modeling. Yet, more importantly, we report that our quality index manages to bring all methods at a comparable level of accuracy, hence decreasing the aligner selection dilemma.

Results

Structural Accuracy Prediction.

The purpose of an alignment reliability index is to discriminate between correctly and incorrectly aligned residues. In order to validate TCS, our novel index, we used two reference collections made of structure-based alignments: BAliBASE 3 (Thompson et al. 2005b) and PREFAB 4 (Edgar 2004a). Both have been designed to benchmark multiple aligners accuracy by comparing target sequence-based MSA with their structure-based reference. Rather than benchmarking relative aligners accuracies, we have used these sets in order to quantify the TCS capacity to discriminate between correctly and incorrectly aligned pairs of residues, regardless of the overall MSA accuracy. For this purpose, we used the *PairTCS* measure (cf Methods). *PairTCS* assigns a score to every pair of aligned residues within an MSA. We estimated its discriminative capacity by quantifying score differences between proven positives (pairs of residues aligned identically to the reference) and proven negatives. We quantified this effect with an AUC measurement using the ROC methodology (cf Methods) originally developed for GUIDANCE.

We first evaluated the effect of the library building protocol (Table 1). We did so using MAFFT because it is the most accurate of the aligners supported by all the evaluation methods compared here. We found the TCS protocol to be more discriminative than any alternative, both on BAliBASE 3 (Thompson et al. 2005b) and PREFAB 4 (Edgar 2004a). TCS is about 4 percentage points more discriminative than GUIDANCE, the second best method, and manages to do so more than 3 times faster. The next best alternative is TCS_FM. TCS_FM has a discriminative capacity comparable to HoT with a CPU cost 5 times lower (about 15 times lower than GUIDANCE). When testing an MSA evaluation scheme, it is important to insure its robustness across both methods and datasets. We estimated the TCS score discriminative capacity on four more multiple aligners: ClustalW and Muscle, two popular progressive aligners as well as SATe and PRANK, two members of a novel generations of phylogeny aware aligners. These aligners were also selected because their average accuracy varies a lot (Table 2), with more than 10 percentage points of spread between the best and the worst aligners on both BAliBASE 3 and PREFAB 4. Regardless of these variations, the TCS discriminative values remain very similar across all considered aligners. On all datasets/aligners combinations, we found TCS to outperform both GUIDANCE and HoT with a majority of reported differences being statistically significant.

We also estimated the usefulness of filtering methods like Gblocks or trimAl that removes entire columns. We found these filtering methods to be poorly informative from a structural point of view (Table 1). We found this trend to be constant across all combinations of aligners (Table 2, detailed AUC distribution in Supp. Figure 1 to 10) and across all ranges of dataset difficulty (Table 3). These observations suggest that identity and indel-based filtering are not suitable criteria for the identification of structurally correct MSA regions. In order to refine our analysis, we calculated the results shown in Table 1 and 2 for each subcategory of both BAliBASE and PREFAB (Supp. Table 2 and 3). On every tested value, we found TCS to significantly outperform HoT and GUIDANCE, regardless of the aligner.

Numerical stability is an important property for an index measure like TCS. One expects a good measure to be constantly trustworthy, regardless of the aligner accuracy or any dataset specificity (i.e., dataset difficulty, dataset size). Table 2, suggests all indexes to be affected by the overall dataset difficulty. AUC values of most indexes are usually lower for BAliBASE whose reference MSAs also seem to be more challenging to aligners. This observation suggests that it may be harder to discriminate between accurate and inaccurate pairs when dealing with remote MSAs of homologues. To further examine this effect we focused our analysis on two reference subsets: the most challenging (RV11 in BAliBASE 3, [0~20%] in PREFAB 4) and the easiest (RV12, [70~100%]) of MAFFT alignment as shown on Table 3. Interestingly, this analysis shows that all methods tend to increase in performance when datasets become easier, but only up to a certain point. For instance we found predictions made on PREFAB 4 [0~20%], the most challenging dataset, to be often less informative than on the slightly easier RV11 dataset. The trend is even stronger when considering the much easier RV12, for which 88.8% of the residues are correctly aligned. When comparing RV11 and RV12, most methods (except trimAl gappyout) return improved predictions. This trend is clearly inverted on the easiest dataset collection (PREFAB 4

[70~100%]) in which most methods - except trimAl - undergo a severe drop in predictive capacities. This may be partly explained by the severe imbalance between correctly and incorrectly aligned residues on these very easy datasets in which 94.2% of the residues are correctly aligned by MAFFT, thus making it more challenging to predict the incorrectly aligned residues. It is noteworthy that even with such a low proportion of misaligned residues, TCS remains the best predictive method. In general, TCS seems to be less affected than HoT or GUIDANCE with respect to the similarity level within a dataset while being significantly more informative than trimAl or Gblocks. We also evaluated the effect of dataset size on prediction performance. To do so we separately calculated the AUC for datasets with less or more than 10 sequences (Supp. Table 1 and Supp. Figure 11 to 15) and found TCS to be more stable than GUIDANCE with respect to dataset size. This may be explained by the GUIDANCE algorithm dependency on the guide tree sampling procedure, causing GUIDANCE AUC to drops when dealing with datasets containing less than 10 sequences.

Confidence in residue level

The AUC is a useful measure to compare the relative performance of various methods, but it does not relate directly to the practical usage of a reliability estimator. Users need a measure that makes it possible to apply automated filtering methods. The *PairTCS* measure is not convenient for this purpose, hence the need for *ResidueTCS* (cf Methods), a measure that assigns a score to every individual residue in an MSA so that poorly aligned residues can be automatically filtered out. In order to do the benchmark, we tagged as correct, all residues correctly aligned to 50% or more of other residues within the same column of the evaluated MSA. This criterion allowed us to define sensitivity as the fraction of correct residues (over the total before filtering) retained after filtering and specificity as the fraction correctly aligned residues (over the total after filtering). Results (Figure 1, Supp. Table 4) suggest that sensitivity and specificity do not depend on an aligner’s overall accuracy and that similar levels are reached on the five considered aligners. Interestingly, specificity that is the capacity to retain a large fraction of correct residues, is the most affected by the aligner's accuracy, and tends to decreases modestly when considering high accuracy aligners like MAFFT or SATe. Using a very stringent cutoff, one can identify nearly half of the correctly aligned residues (43.0%) with a reliability of 94.3%. As an alternative, the best trade-off between sensitivity and specificity is obtained when keeping all residues having a score superior to 0.6, in which case we obtain an average specificity of 80.4% for an average sensitivity of 73.9%. These values do not vary much between SATe, the most accurate aligner tested here (Spec. 77.5%, Sens. 74.2%) and ClustalW, the less accurate (Spec. 82.5%, Sens. 74.8%). This important result suggests that when using the right evaluation metrics, high structural confidence can be established on more than a third of the columns in an MSA.

Discrimination between two alternative MSAs

Residue scores are very useful when doing high quality modeling and other kind of fine grain analysis. In other situations, typically when running large-scale pipeline analysis, one is often more concerned with comparing alternative MSAs, or deciding objectively if an MSA is good enough for database inclusion. For this purpose, one needs a global MSA score allowing qualitative comparisons like the *AlignmentTCS* that estimates an MSA global TCS score. The usefulness of such a score is to allow the effective discrimination between two alternative MSAs of the same sequences. We validated this metrics using the strategy developed in for STRIKE (Kemena et al. 2011). It involves generating MSAs with several methods, measuring the reference score for each MSA and the *AlignmentTCS* score, and comparing these scores on every possible pair of alternative MSAs (i.e. same sequences different aligner). When doing so, one measures the difference on the reference structure based score and the difference on the TCS metrics. If both differences have the same algebraic sign, it means the metrics agree on the relative ranking of the two alternative MSAs. A graphic representation is displayed on Figure 2, where all comparisons having the same algebraic sign appear as dots in the top right and the bottom left quadrants. We quantified this effect and found (Table 4) that on BAliBASE, 83.5% of the points land in the correct quadrants, as compared to 71.1% when doing similar analysis with GUIDANCE. Albeit less good, the readout on PREFAB also gave more than 10 percentage points of spread (72.5% vs. 60.5%).

Phylogenetic Reconstruction

It is tempting to believe that high accuracy MSAs should result in more accurate phylogenetic reconstruction. We decided to address this important question by exploring the TCS index capacity of improving phylogenetic reconstruction accuracy. Several methods have been designed for this purpose. They rely on the same principle: poor columns are identified and filtered out in an attempt to improve the subsequent phylogeny. This principle is used in Gblocks (Castresana 2000; Talavera and Castresana 2007) and trimAl (Capella-Gutierrez et al. 2009), a more recent follow-up. The main weakness of such protocols is the arbitrary cutoff between retained and removed columns. The other main weakness is their reliance on sequence identity and their tendency to exclude phylogenetically informative sites containing too many indels. By contrast the TCS method does not explicitly rely on sequence identity but rather on alignment robustness. We designed two protocols: Filtered TCS, that uses the *ColumnTCS* in order to remove all columns having a score lower than 2 (cf Methods) and Weighted Replicate TCS, a replicative scheme where each column is replicated a number of time proportional to its *ColumnTCS* score. This procedure has the advantage of remaining entirely compatible with all MSA based phylogenetic reconstruction methods, including bootstrap estimate procedures.

Simulation data

We first ran this analysis on ROSE-generated (Stoye et al. 1998) simulated datasets previously used to validate Gblocks and trimAl. ROSE uses a pre-defined tree topology to generate a set of sequences whose mutation patterns are compatible (sampling noise included) with the evolutionary scenario (tree). One can then evaluate the performance of a tree reconstruction method by estimating its capacity to reproduce the source tree, for instance by comparing normalized Robinson and Foulds (RF) topological distances (Figure 3, Supp. File: phylogenetic_simulation_MAFFT.csv) or by estimating the normalized fraction of missing (FN) and supported incorrect branches (FPire) (Supp. Figure 24 to 32).

When running ML on MAFFT alignments with both the weighted replicate TCS and the filtered TCS, we found the weighted TCS protocols to be the best overall performing method, an observation that was confirmed when running similar analysis on ClustalW and ProbCons MSAs (Supp. Figure 16 to 17, Supp. Files: phylogenetic_simulation_ClustalW.csv, phylogenetic_simulation_ProbCons.csv). Results measured on the topologically less accurate NJ or MP trees were more ambiguous (Supp. Figure 18 to 23). While RF analysis suggests a moderate dominance of weighted methods, normalized FN+FPire readouts gave a conflicting readout suggesting an slight edge for filtering methods (Gblocks and trimAl) used in strict mode (Supp. Figure 24 to 32) even though the higher tendency of these methods to generate more unresolved branch nodes may explain part of this observation (Supp. Figure 33 to 41). Both metrics, however, agreed on the ranking of weighted TCS as the best overall method for NJ, MP and ML tree reconstruction on short datasets (400 sites), regardless of the aligner.

The filtering and weighting protocols outlined for TCS are not restricted to our metrics and can easily be deployed in combination with any numerical reliability index such as GUIDANCE or HoT. We therefore quantified GUIDANCE and HoT accuracy on these same datasets, both as filtered or weighted indexes. The filtering threshold was estimated by considering the value giving the best readout on the considered datasets. As one can see on Figure 3, TCS is consistently more accurate than HoT or

GUIDANCE. More interestingly, the ML reconstructions also suggest the overall superiority of the weighted index over the filtered, regardless of the underlying index.

Empirical Yeast data

The main limitation of simulated datasets is their artificial nature. They rely on simplistic evolutionary models that cannot be directly validated. For a meaningful biological benchmark to be carried out, one would need the phylogenetic equivalent of structure based reference alignments. We tried to assemble such a dataset starting from collection of yeast 1-to-1 orthologous genes compiled on 7 species (Wong et al. 2008) and enriching it into datasets whose tree topologies appear to support the established subtree of the Yeast Tree of Life (subToL) (Rokas et al. 2003). This tree is very well supported by a number of subsequent re-analyses (Phillips et al. 2004; Taylor and Piel 2004; Ren et al. 2005; Burleigh et al. 2006; Ane et al. 2007; Criscuolo and Gribaldo 2010). Our selection, carried out on the basis of the ML trees established on 7 different aligners (cf Methods) can only be an enrichment, as many confounding factors play a role on topology congruence or incongruence, including incomplete lineage sorting, lateral transfer and spurious orthologous assignments. Yet, provided the enrichment is high enough, such a dataset can then be used to compare aligners. This procedure resulted in a collection of 853 sets of orthologous datasets.

We first estimated the filtered TCS measure on the four aligners and found the best filtering value (i.e. the threshold yielding the lowest RF score and the largest number of true topologies) to be relatively consistent across aligners. We then applied the 6 other filtering procedures and found filtered TCS to be systematically more accurate than the other filtering metrics (Table 6, top rows), regardless of the considered MSA method. We finally estimated the relative accuracy of the weighted version of TCS, HoT, GUIDANCE and trimAl in comparison to their filtered counterparts (Table 6, bottom rows). We found -

in agreement with simulations - that weighted replicates systematically lead to higher accuracy trees, regardless of the considered index (HoT, GUIDANCE, trimAl or TCS), with TCS being significantly more accurate than other weighting schemes.

Overall, our finding indicates that TCS is the most suitable method. At worst it does not degrade the topologies (as the other filtering methods tend to do, especially the stringent ones), and at best, it results in significant topological improvements that bring most methods to a comparable level of accuracy, close to the average of 661 ToL-like topologies. It is remarkable that when doing Maximum Likelihood tree reconstruction, methods relying on columns removal using conservation as a removal criterion (Gblocks and trimAl) almost systematically induce a decrease in accuracy over the original unfiltered MSAs. This trend is dramatically amplified when considering the most stringent setups (Gblocks stringent and trimAl strictplus), an observation in agreement with Liu *et. al.* 's report who indicated column filtering not to be a source of improvement when benchmarking SATe (Liu et al. 2009). Of course, the analysis of this last benchmark is hampered by our ignorance of the exact portion of datasets effectively supporting a ToL topology. The strength of our results lies therefore mostly in the readout consistency across the tested aligners, and in the relative agreement between structure based and evolutionary validations.

Discussion and Conclusion

In this study, we describe the benchmarking of TCS, a novel MSA reliability index that relies on the T-Coffee algorithm. Given a set of sequences, a target MSA computed with any third party method, and all possible pairwise comparisons of these same sequences (library), the TCS index uses a consistency transformation in order to assign a reliability index (normalized between 0 and 1), to every pair of aligned residues in the target MSA, to every aligned residue, to every column and to the whole alignment. In the first section of the study, we show the TCS to be highly informative with respect to structural accuracy prediction. On BAliBASE 3 (Thompson et al. 2005b), it identifies more than 40% of the correctly aligned residues, with reliability higher than 94%. We also show TCS predictions to be almost unaffected by aligners differences in accuracy. In our study, the TCS discriminative capacity is similar on MAFFT, SATe and ClustalW MSAs, even though these aligners have different accuracies. TCS is more informative than GUIDANCE and HoT, the two most popular alternatives. CPU-wise, TCS is about 3 times faster than GUIDANCE and comparable to HoT. Our validation confirms the superiority of the TCS measure at all levels: better discrimination between correctly and incorrectly aligned pair residues, good identification of correctly aligned residues and better discrimination between alternative MSAs of the same sequences.

Since the TCS metrics also returns a column score, we evaluated its capacity to help improving phylogenetic reconstruction by using it as a site-weighting scheme. While most protocols involve removing potentially spurious columns (typically those containing a lot of gaps), ours was designed so as to enhance the most reliable positions by replicating columns according to their TCS score within an MSA. This protocol has two main advantages: it is entirely parameter-free and retains all the original information, including confounding regions of the original MSA. Such regions being usually the consequence of local evolutionary abnormalities, it certainly makes sense to retain this information so that it can be reflected in the final tree branch lengths and bootstrap confidence. We did validate this procedure on two datasets: an artificial one, using a popular strategy based on ROSE and another one based on an empirical dataset built for this study. Both analyses confirm the superiority of our expanded MSAs, regardless of the method used to generate the original MSA.

Interestingly, the trends uncovered on the simulated phylogenetic datasets are significantly stronger in the empirical one, even though this dataset is made of very small trees (7 tips). The empirical dataset nonetheless recapitulates most of the results recently published in the literature, including the superiority of phylogeny aware aligners and the limited usefulness of gap trimming when building phylogenies. Altogether, it suggests that empirical datasets could be a useful addition when benchmarking tree reconstruction methods, where they could play a role similar to that of structure-based MSA collections. Such an approach would make it easier to separate the aspect of phylogenetic reconstruction that relies on a pure mathematical optimization, and therefore require simulation, from the less well-established aspect relating to the biological relevance of alternative protocols.

The main result of this study is probably its contribution to the debate on the existence of some continuity between phylogeny reconstruction and structural comparison. Our results are somehow ambivalent. On the one hand, our work confirms that the same metrics can be used to recognize in an MSA both the structurally correct portions, and those most likely to support a correct phylogeny. SATe's readout, the best method on both systems, supports this interpretation. On the other hand, we found that PRANK, the other aligner performing best on phylogenetic reconstruction, is also one of those having the lowest performances on BAliBASE and PREFAB. This suggests some potential discontinuity between phylogenetically informative MSAs and accurate homology based structural modeling (Blackburne and Whelan 2013). It might be explained by the high redundancy of signal when doing phylogenetic reconstruction since in theory; each site (column) potentially contains the true history of the considered family. Properly aligning a fraction large enough may therefore allow correct tree estimation. Structural validation is, by contrast much more stringent as it requires all considered sites to be correctly aligned. In short, having a good aligner may be less critical when doing phylogenetic reconstruction. The TCS, however, appears to be little affected by this. On the one hand, it can recognize in PRANK's MSA the most structurally trustworthy regions and on the other hand, its non-destructive MSA processing for phylogenetic reconstruction allows it to improve the tree reconstruction potential of most aligners. TCS is an extremely versatile protocol, lending itself to an infinite number of variations, thanks to its reliance on third party libraries. We have no doubt that future work will bring forward novel applications of consistency for the estimation of biological models.

Methods

Transitive Consistency Score

The transitive consistency score (TCS) is an extended version of the T-Coffee scoring scheme. It departs from the original CORE score in the normalization (Notredame and Abergel 2003). Given a library of pairwise alignments for the set of sequence *S*, this score is used to estimate the score of aligning two residues R_i^x (i^{th} residue from sequence *x*) and R_j^y (j^{th} residue of sequence *y*) by identifying all intermediate residues R_k^z from a third sequence *z* that connects R_i^x and R_j^y through the two following pairwise alignments: $R_i^x R_k^z$ and $R_k^z R_j^y$. Given the entire pairwise library, the reliability score is first defined as the sum of all $R_i^x R_j^y$ pairs weights linked through all possible R_k^z residues, with each R_k^z residue contributing *Min* ($R_i^x R_k^z$, $R_k^z R_j^y$) to the final score. This score is then normalized by its upper bound, estimated by considering the maximum score over all possible pair combinations involving R_i^x or/and R_j^y through an intermediate compatible R_k^z:

$$PairTCS\left(R_i^x, R_j^y\right) = 2\,\frac{\sum_z^S Min\left(R_i^x R_k^z, R_k^z R_j^y\right)}{\sum_z^S Min\left(R_i^x R_k^z, R_k^z R_*^y\right) + \sum_z^S Min\left(R_*^x R_k^z, R_k^z R_j^y\right)} \quad (1)$$

, where * denotes any residue, including R_i^x or R_j^y (i.e. the direct alignment of R_i^x with R_j^y). Measure (1) defines the score of aligned residue pairs scaled between 0 and 1. $PairTCS\left(R_i^x, R_j^y\right)$ is 0 when both pairwise residue alignments $R_i^x R_k^z$ and $R_k^z R_j^y$ do not exist in the library for any intermediate residues R_k^z. In order to define a column score, this same score is averaged across all aligned residues within the considered column. In the formula below, C_i^x is a residue in column *i* of sequence *x* (excluding gaps) and C_i is the list of residues in that same columns (excluding gaps):

$$ColumnTCS(C_i) = \frac{\sum_x^{|C_i|}\sum_{y\neq x}^{|C_i|} PairTCS(C_i^x, C_i^y)}{|C_i|*(|C_i|-1)} \quad (2)$$

Using the formulation developed in (Notredame and Abergel 2003) for the CORE index, we defined the reliability index of every individual aligned residue by averaging the TCS score over every pair of residues:

$$ResidueTCS(C_i^x) = \frac{\sum_{y\neq x}^{|C_i|} PairTCS(C_i^x, C_i^y)}{|C_i|-1} \quad (3)$$

All these values can then be combined into one unique index for the whole alignment *A*, as shown here with $L_x$being the length of sequence *x*:

$$AlignmentTCS(A) = \frac{\sum_x^{|S|}\sum_i^{Lx} ResidueTCS(C_i^x)}{\sum_x^{|S|} L_x} \quad (4)$$

A major strength of this formulation is its independence from the library generation procedure and the possibility to compute it regardless of the source of $R_i^x R_j^y$. Any reasonable source of alignment can be used to populate this library. In practice we used three protocols, listed in the next section.

TCS Evaluation Libraries

The TCS can be used to evaluate any MSA. It only requires a library of pre-computed alternative alignments, pairwise or multiple. In practice, one is entirely free to define this library in any suitable way. In the context of this work, we have used the three following protocols:

- *TCS_original*. This protocol corresponds to the original T-Coffee. It involves computing all pairwise alignments using ClustalW (Thompson et al. 1994) and Lalign (Huang and Miller 1991). In this library, pairs of residues are weighted by the average identity measured on the pairwise alignment they were extracted from. When different alignments contribute the same pair, the final weight is the maximum of the alternative values. The T-Coffee command used to generate these libraries is:

t_coffee –seq <seq_file> -method clustalw_pair, lalign_id_pair –out_lib <library> -lib_only

- *TCS*. This protocol corresponds to the current default T-Coffee whose libraries are populated using the ProbCons pair-HMM (Do et al. 2005). These libraries only contain residue pairs whose posterior probability of being aligned are higher than 0.99. The primary weights are set to these values. Libraries are computed using the following command:

  t_coffee –seq <seq_file> -method proba_pair –out_lib <library> -lib_only

- *TCS_FM.* This protocol uses the procedure developed for the Ensembl Compara pipeline (Flicek et al. 2010) and relies on generating MSAs using fast aligners: MAFFT (Katoh et al. 2002), MUSCLE (Edgar 2004b) and Kalign (Lassmann and Sonnhammer 2005). These MSAs are then used to extracts all the pairwise projections and populate the library in a standard way. Libraries are computed using the following command:

  t_coffee –seq <seq_file> -method kafft_msa,kalign_msa,muscle_msa –out_lib <library> -lib_only

TCS is available both in command line and as a web-server with default TCS.

TCS Evaluation Procedure

The above libraries were used to evaluate MSAs produced with the most commonly used multiple aligners by default setting, including: ClustalW 2.1 (Larkin et al. 2007), MAFFT 6.711 with FFT-NS-2 model (Katoh and Toh 2008) and MUSCLE 3.8.31 (Edgar 2004a). We also used two recent phylogeny aware aligners: PRANK v.100802 (Loytynoja and Goldman 2008) and SATe 2.2.5 (default setting: MAFFT for alignment, RAxML for tree estimation) (Liu et al. 2012). T-Coffee, ProbCons and related consistency based aligners were voluntarily excluded because they all are heuristics explicitly designed to optimize objective functions very similar to TCS. Using them would have meant simultaneously estimating the TCS reliability and the optimization capacity of these heuristics, thus potentially resulting in confounding effects. MSAs were evaluated using the following command:

```
t_coffee –infile=<target_MSA> –evaluate –lib <library> -output \
sp_ascii,score_ascii,score_html,score_pdf,tcs_column_filter2,tcs_weighted,tcs_replicate100
```

- *sp_ascii* is a format reporting the TCS score of every aligned pair (*PairTCS*) in the target MSA.
- *score_ascii* reports the average score of every individual residue (*ResidueTCS*) along with the average score of every column (*ColumnTCS*) *and* the global MSA score (*AlignmentTCS*).
- *score_html* score_ascii in html format with color code (Figure 4).
- *score_pdf* will transfer score_html into pdf format.
- *tcs_column_filter2* outputs an MSA in which columns having *ColumnTCS* lower than 2 are removed.
- *tcs_weighted* outputs an MSA in which columns are duplicated according to their *ColumnTCS* weight.
- *tcs_replicate100* outputs 100 replicate MSAs in which columns are randomly drawn according to their weights (*ColumnTCS*).

Structural Reference Datasets

Two amino acid dataset collections were used to estimate structural correctness: BAliBASE 3 (Thompson et al. 2005b) that contains 218 sets classified in 5 categories and PREFAB 4 (Edgar 2004a), a collection of homologous structure pairs embedded in 50 homologous sequences. Accuracy estimates were done using the core regions (as defined by the authors) in terms of the average similarity between evaluated MSAs and their references measured as the fraction of identical pairs (Sum-of-Pairs). Fine-grain benchmarking was carried out by extracting all core regions (as defined in the databases) in the reference alignment as list of residue pairs.

Structural Benchmarking

TCS, like HoT or GUIDANCE, makes it possible to systematically evaluate the score of each pair of aligned residues and check the relation between score-based ranking and structural correctness. We did a Receiving Operator Curve (ROC) analysis using the validation procedure reported in GUIDANCE. It involves extracting from the target MSA all residue pairwise alignment projections, keeping only those containing at least one core region residue and labeling these pairs according to the reference: Proven Positives (PP) for those found in the reference MSA, Proven Negatives (PN) for the others. These same pairs were then evaluated using either the TCS, GUIDANCE or HoT procedure (default) and sorted according to their scores. The list of labeled ordered pairs was then used to do a ROC analysis so as to estimate the Area Under Curve (AUC) with the ROCR R package (Sing et al. 2005). AUC values were used to compare performances for each subset of BAliBASE 3 and PREFAB 4. We also evaluated the structural relevance of trimAl and Gblocks filtering. For this purpose, we simply assigned a score of 0 to all filtered residues and a score of 1 to the remaining ones, thus allowing a validation comparable to that carried out on TCS, HoT or GUIDANCE.

Phylogenetic Reference datasets

Simulated. We used the Gblocks (Talavera and Castresana 2007) simulated amino acid dataset (16 tips) and its follow up, trimAl (Capella-Gutierrez et al. 2009) (32 and 64 tips). Both were generated using ROSE (Stoye et al. 1998). We only used the asymmetric mode as it has been reported to be the most challenging. Alignments were constructed using ClustalW, MAFFT and ProbCons.

Empirical. We used the Wong *et al.* (Wong et al. 2008) dataset, made of 1502 clusters of 7 orthologous gathered in 7 yeast genomes. Wong used this dataset to compare the ML (PAUP) phylogenies resulting from using 7 aligners (DCA, ClustalW, Dalign, MAFFT, Muscle, ProbCons and T-Coffee). There are 1494 datasets in which at least one aligner produces significantly similar tree to yeast ToL by Shimodaira-Hasegawa test (Shimodaira and Hasegawa 1999) (Supp. File: phylogenetic_yeast_SHtest.csv). We conservatively select the 853 datasets in which at least one aligner yields a phylogeny topology identical to the canonical yeast ToL (Rokas et al. 2003) so as to define an empirical dataset enriched for the ToL topology.

Alignment post processing - filtering

Phylogenetic benchmarking was carried out in order to quantify the effect of MSA filtering and weighting. We tested the two most popular procedures: Gblocks and trimAl. Both work along the same principle that involves filtering out positions in an MSA on the basis of its conservation. For the sake of comparison we included Gblocks using the *stringent* and the *relaxed* procedures that keeps all positions containing less than 50% of gapped positions and we also used trimAl that was benchmarked in two modes: *gappyout,* which automatically selects gap cut-off score depending on MSA's gap distribution and *strictplus,* which automatically selects block size. For testing the usefulness of HoT and GUIDANCE score on phylogenetic reconstruction, filtered MSAs generated by GUIDANCE and HoT (Without_low_SP_Col) are also included. MSA columns were removed if their confidence scores are below default cutoff (0.93). The filtered version of TCS was validated using a cutoff of 2 that yields the best readout on the yeast reference dataset (Table 5).

Alignment post processing - weighting

When doing phylogenetic validation, we defined an alternative to the filtering protocols called weighted replication. In this protocol, the indexes (HoT, GUIDANCE, trimAl or TCS) were used to amplify reliable columns within each MSA. This process is achieved by outputting a re-coded MSA in which each column is represented a number of time equal to its $ColumnTCS(C_i)$ *10, with $ColumnTCS(C_i)$ normalized within a 1~10 range (i.e. no column is deleted). In trimAl, replication was done according to trimAl gap score (the third column outputted by "-sgc" option; = 1 – gap percentage), in HoT with the column score and in GUIDANCE with the Guidance_col_col.scr. No weighting scheme was available for Gblocks.

Phylogenetic Benchmark

Filtered MSAs were then used to estimate Neighbor Joining (NJ), Maximum Parsimony (MP) and Maximum Likelihood (ML) trees. For simulated set, the procedure used to infer NJ, MP and ML follows Gblocks's publication (Talavera and Castresana 2007). NJ were build by *Neighbor* of Phylip on the pairwise protein distance calculated by *Protdist* of the same package on the Jones-Taylor-Thornton (JTT) model of protein evolution (Jones et al. 1992). MP was build by *Protpars* of Phylip with 50 random initializations to insure a thorough tree search. ML was build by PhyML version 3.0 (Guindon et al. 2010) with the JTT model, default four rate categories and the Gamma distribution estimated by the ML of the phylogeny. For empirical set, the procedure used to infer ML follows Wong et al. publication (Wong et al. 2008). ML is build by PAUP version 4.10b (Swofford 2003) under the GTR+γ model of DNA substitution with four rate categories.

Topological Error Measures

Trees were compared to the references using treedist of Phylip package version 3.68 (Felsenstein 1989) implementation of the Robinson-Foulds (RF) topological distance measure (Robinson and Foulds 1981), which ranges from 0 to 2*n*-6 given *n* species (normalized RF = RF / 2n-6). Taking into account known RF limitations when comparing trees (Hartmann and Vision 2008) we further decomposed it into two metrics, *FP* and *FN*, respectively. *FP* is the number of model branches that do not appear in the reference tree, and *FN* the number of branches within the reference that are not found in the model tree (Desper and Gascuel 2004). We also took into account the observation that branches not supported by any substitution cannot be recovered, except by chance (Desper and Gascuel 2004) and defined as *I*(*T*), the number of non-supported branches with length smaller than L^{-1}, where *L* is MSA length. The *I*(*T*) measure could not be used for the reference tree and MP model trees that do not contain un-resolved nodes, it was estimated for all the other trees using *CompareTree.pl* in “Fast Tree-Comparison Tools” (Price et al. 2009, 2010). So, *I*(*T*) is analyzed for ML and NJ model trees.

Computation

MSAs and trees were estimated using the Amazon elastic cloud (five cc2.8xlarge instances in 255 hours).

Acknowledgements

The authors thank Eyal Privman for helping benchmark HoT and GUIDANCE, and Jose Castresana and Salvador Capella for providing the data necessary for the phylogenetic validation benchmark, and Tal Pupko for his many advices on revising the manuscript. We also thank the two anonymous reviewers for the amount of time and reflection they have spent on this manuscript, with so many demanding yet highly constructive comments.

Funding: This work was supported by the Spanish Ministry of Economy and Competitiveness (grant Nr. BFU2011-28575); the Secretariat of Universities and Research, Dept. of Economy and Knowledge of the Government of Catalonia (2009 SGR 951); the "la Caixa" International Fellowship Programme for a pre-doctoral fellowship at the CRG (to J.M.C.) and the European Research Council (ERC-2008-AdG No 232947 to J.M.C.). We acknowledge support from the Spanish Ministry of Economy and Competitiveness, 'Centro de Excelencia Severo Ochoa 2013-2017', SEV-2012-0208.

References

Ane C, Larget B, Baum DA, Smith SD and Rokas A. 2007. Bayesian estimation of concordance among gene trees. Mol Biol Evol 24: 412-426.

Blackburne BP and Whelan S. 2013. Class of multiple sequence alignment algorithm affects genomic analysis. Mol Biol Evol 30: 642-653.

Burleigh JG, Driskell AC and Sanderson MJ. 2006. Supertree bootstrapping methods for assessing phylogenetic variation among genes in genome-scale data sets. Syst Biol 55: 426-440.

Capella-Gutierrez S, Silla-Martinez JM and Gabaldon T. 2009. trimAl: a tool for automated alignment trimming in large-scale phylogenetic analyses. BIOINFORMATICS 25: 1972-1973.

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

Claude JB, Suhre K, Notredame C, Claverie JM and Abergel C. 2004. CaspR: a web server for automated molecular replacement using homology modelling. Nucleic Acids Res 32: W606-609.

Criscuolo A and Gribaldo S. 2010. BMGE (Block Mapping and Gathering with Entropy): a new software for selection of phylogenetic informative regions from multiple sequence alignments. BMC Evol Biol 10: 210.

Desper R and Gascuel O. 2004. Theoretical foundation of the balanced minimum evolution method of phylogenetic inference and its relationship to weighted least-squares tree fitting. Mol Biol Evol 21: 587-598.

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

Edgar RC. 2004a. MUSCLE: a multiple sequence alignment method with reduced time and space complexity. BMC Bioinformatics 5: 113.

Edgar RC. 2004b. MUSCLE: multiple sequence alignment with high accuracy and high throughput. Nucleic Acids Res 32: 1792-1797.

Felsenstein J. 1989. PHYLIP-Phylogeny Inference Package(Version 3.2). In. Cladistics. p. 164-166.

Flicek P, Aken BL, Ballester B, Beal K, Bragin E, Brent S, Chen Y, Clapham P, Coates G, Fairley S, et al. 2010. Ensembl's 10th year. Nucleic Acids Res 38: D557-562.

Guindon S, Dufayard JF, Lefort V, Anisimova M, Hordijk W and Gascuel O. 2010. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of PhyML 3.0. Syst Biol 59: 307-321.

Hartmann S and Vision TJ. 2008. Using ESTs for phylogenomics: can one accurately infer a phylogenetic tree from a gappy alignment? BMC Evol Biol 8: 95.

Huang X and Miller W. 1991. A time-efficient, linear-space local similarity algorithm. Advances in Applied Mathematics 12: 337-357.

Jones DT, Taylor WR and Thornton JM. 1992. The rapid generation of mutation data matrices from protein sequences. Computer applications in the biosciences : CABIOS 8: 275-282.

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

Katoh K and Toh H. 2008. Recent developments in the MAFFT multiple sequence alignment program. Brief Bioinform 9: 286 - 298.

Kemena C and Notredame C. 2009. Upcoming challenges for multiple sequence alignment methods in the high-throughput era. Bioinformatics 25: 2455-2465.

Kemena C, Taly JF, Kleinjung J and Notredame C. 2011. STRIKE: evaluation of protein MSAs using a single 3D structure. BIOINFORMATICS 27: 3385-3391.

Kim J and Ma J. 2011. PSAR: measuring multiple sequence alignment reliability by probabilistic sampling. Nucleic Acids Res 39: 6359-6368.

Landan G and Graur D. 2007. Heads or tails: a simple reliability check for multiple sequence alignments. Mol Biol Evol 24: 1380-1383.

Landan G and Graur D. 2008. Local reliability measures from sets of co-optimal multiple sequence alignments. Pac Symp Biocomput: 15-24.

Larkin M, Blackshields G, Brown N, Chenna R, Mcgettigan P, Mcwilliam H, Valentin F, Wallace I, Wilm A, Lopez R, et al. 2007. Clustal W and Clustal X version 2.0. BIOINFORMATICS 23: 2947 - 2948.

Lassmann T and Sonnhammer EL. 2005. Kalign--an accurate and fast multiple sequence alignment algorithm. BMC Bioinformatics 6: 298.

Lipman DJ, Altschul SF and Kececioglu JD. 1989. A tool for multiple sequence alignment. Proc Natl Acad Sci U S A 86: 4412-4415.

Liu K, Raghavan S, Nelesen S, Linder CR and Warnow T. 2009. Rapid and accurate large-scale coestimation of sequence alignments and phylogenetic trees. Science 324: 1561-1564.

Liu K, Warnow TJ, Holder MT, Nelesen SM, Yu J, Stamatakis AP and Linder CR. 2012. SATe-II: very fast and accurate simultaneous estimation of multiple sequence alignments and phylogenetic trees. Syst Biol 61: 90-106.

Liu Y, Schmidt B and Maskell DL. 2010. MSAProbs: multiple sequence alignment based on pair hidden Markov models and partition function posterior probabilities. BIOINFORMATICS 26: 1958-1964.

Loytynoja A and Goldman N. 2008. Phylogeny-aware gap placement prevents errors in sequence alignment and evolutionary analysis. Science 320: 1632-1635.

Needleman SB and Wunsch CD. 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 and Abergel C. 2003. Using multiple alignment methods to assess the quality of genomic data analysis. Bioinformatics and Genomes: Current Perspectives Horizon Scientific Press, Wymondham, UK: 30-50.

Notredame C, Higgins DG and Heringa J. 2000. T-Coffee: A novel method for fast and accurate multiple sequence alignment. J Mol Biol 302: 205-217.

Notredame C, Holm L and Higgins DG. 1998. COFFEE: an objective function for multiple sequence alignments. BIOINFORMATICS 14: 407-422.

Penn O, Privman E, Landan G, Graur D and Pupko T. 2010. An alignment confidence score capturing robustness to guide tree uncertainty. Mol Biol Evol 27: 1759-1767.

Phillips MJ, Delsuc F and Penny D. 2004. Genome-scale phylogeny and the detection of systematic biases. Mol Biol Evol 21: 1455-1458.

Price MN, Dehal PS and Arkin AP. 2009. FastTree: computing large minimum evolution trees with profiles instead of a distance matrix. Mol Biol Evol 26: 1641-1650.

Price MN, Dehal PS and Arkin AP. 2010. FastTree 2--approximately maximum-likelihood trees for large alignments. PLoS One 5: e9490.

Ren F, Tanaka H and Yang Z. 2005. An empirical examination of the utility of codon-substitution models in phylogeny reconstruction. Syst Biol 54: 808-818.

Robinson D and Foulds L. 1981. Comparison of phylogenetic trees. Math Biosci 53: 131 - 147.

Rokas A, Williams BL, King N and Carroll SB. 2003. Genome-scale approaches to resolving incongruence in molecular phylogenies. Nature 425: 798-804.

Roshan U and Livesay DR. 2006. Probalign: multiple sequence alignment using partition function posterior probabilities. BIOINFORMATICS 22: 2715-2721.

Rost B. 1999. Twilight zone of protein sequence alignments. Protein Eng 12: 85-94.

Sander C and Schneider R. 1991. Database of homology-derived protein structures and the structural meaning of sequence alignment. Proteins 9: 56-68.

Shimodaira H and Hasegawa M. 1999. Multiple comparisons of log-likelihoods with applications to phylogenetic inference. Molecular biology and evolution 16: 1114-1116.

Sing T, Sander O, Beerenwinkel N and Lengauer T. 2005. ROCR: visualizing classifier performance in R. Bioinformatics 21: 3940-3941.

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

Swofford DL. 2003. PAUP*. Phylogenetic Analysis Using Parsimony (*and Other Methods). Version Version 4. Sunderland, Massachusetts: Sinauer Associates.

Talavera G and Castresana J. 2007. Improvement of phylogenies after removing divergent and ambiguously aligned blocks from protein sequence alignments. Syst Biol 56: 564-577.

Taylor DJ and Piel WH. 2004. An assessment of accuracy, error, and conflict with support values from genome-scale phylogenetic data. Mol Biol Evol 21: 1534-1537.

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

Thompson JD, Higgins DG and Gibson TJ. 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 JD, Koehl P, Ripp R and Poch O. 2005b. BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. Proteins 61: 127-136.

Wang L and Jiang T. 1994. On the complexity of multiple sequence alignment. J Comput Biol 1: 337-348.

Wong KM, Suchard MA and Huelsenbeck JP. 2008. Alignment uncertainty and genomic analysis. Science 319: 473-476.

Tables

Table 1. Average AUC (%) for structural correctness predictions as measured using TCS with different library protocols, HoT, GUIDANCE, Gblocks and trimAl on MAFFT alignment. The last column (Time) indicates the CPU time in seconds, measured on BAliBASE 3.

	BAliBASE	PREFAB	Time (s)
library protocols			
TCS	**94.44**	**89.24**	17,244
TCS_original	91.20	83.83	43,258
TCS_FM	87.28	80.03	3,093
GUIDANCE	90.28	85.74	66,368
HoT	82.66	80.30	16,449
Gblocks relax	64.56	60.99	3*
Gblocks stringent	61.91	59.49	4*
trimAl gappyout	52.38	54.26	2*
trimAl strictplus	60.64	61.66	6*

*Running time is only calculated for filtering MSA part.

Table 2. Average AUC (%) for structural correctness prediction as measured using TCS, HoT, GUIDANCE, Gblocks and trimAl. SPs denotes the average similarity between evaluated MSAs and their references measured as the fraction of identical pairs (Sum-of-Pairs). The best performance for each aligner is marked in bold. Entries with "-" indicate measurements that could not be carried out for a lack of support of the considered method for the corresponding aligner. Measurements significantly better than all others in the same column are shown in italics (Wilcoxon Signed-Rank Test in 0.05 significance level, by R *wilcoxon.test* function: paired = TRUE, alternative = "greater").

	ClustalW	MAFFT	Muscle	PRANK	SATe
BAliBASE					
SPs	0.714	0.807	0.793	0.765	0.831
TCS	***96.46***	***94.44***	***94.51***	***96.93***	***93.25***
HoT	90.95	82.66	-	-*	-
GUIDANCE	87.69	90.28	92.10	91.68	-
Gblocks relax	62.82	64.56	62.98	65.07	64.02
Gblocks stringent	60.80	61.91	61.65	60.94	62.29
trimAl gappyout	51.50	52.38	51.45	52.64	51.63
trimAl strictplus	59.01	60.64	59.35	64.52	60.84
PREFAB					
SPs	0.595	0.661	0.649	0.614	0.686
TCS	***90.81***	***89.24***	***87.96***	***92.31***	***86.77***
HoT	83.94	80.30	-	-*	-
GUIDANCE	80.64	85.74	85.60	87.34	-
Gblocks relax	61.10	60.99	60.66	67.35	60.35
Gblocks stringent	59.58	59.49	59.16	64.27	59.21
trimAl gappyout	52.74	54.26	52.47	61.29	53.44
trimAl strictplus	60.67	61.66	61.02	67.90	61.24

*Although HoT support the PRANK aligner, there is running error during test.

Table 3. Average AUC (%) for structural correctness prediction as measured on extreme reference datasets. SPs denotes MAFFT MSAs accuracy, measured in Sum-of-Pairs. RV11 and RV12 are from BAliBASE 3. [0~20%] and [70~100%] are from PREFAB 4.

	difficult		easy	
	RV11	[0~20%]	RV12	[70~100%]
SPs	0.536	0.465	0.888	0.942
TCS	91.11	87.16	96.83	78.98
HoT	72.63	81.35	78.79	57.96
GUIDANCE	83.51	86.03	92.64	62.01
Gblocks relax	60.65	57.56	73.28	62.78
Gblocks stringent	57.10	55.70	73.40	59.23
trimAl gappyout	53.02	52.64	51.88	60.01
trimAl strictplus	57.40	57.94	65.47	65.61

Table 4. Relative score reliability. Each dataset was aligned with ClustalW, Muscle and MAFFT and evaluated with the corresponding method (TCS, TCS_original, TCS_FM or GUIDANCE) as well as Baliscore (BAliBASE) or qscore (PREFAB). The entries indicate the fraction of pairwise comparison between alternative alignments for which there is agreement in ranking between the considered evaluation method and the structure based evaluation. The #comp entries represent the corresponding number of pairwise comparisons. The best performances are marked in bold.

	BAliBASE 3							PREFAB 4				
	RV11	RV12	RV20	RV30	RV40	RV50	all	0~20	20~40	40~70	70~100	all
# comp.	228	264	246	180	294	96	1,308	2,391	1,962	345	294	4,992
TCS	**82.4**	**87.0**	81.7	85.6	80.6	86.5	**83.5**	**71.7**	**74.8**	67.9	62.7	**72.5**
TCS_original	69.2	84.0	**87.4**	**91.7**	**82.0**	**90.6**	83.1	62.8	71.2	**68.3**	73.6	66.8
TCS_FM	67.4	70.6	70.3	70.0	70.7	69.8	69.9	65.2	70.7	62.6	**85.5**	67.8
GUIDANCE	68.3	73.7	64.2	77.8	72.1	72.9	71.1	59.9	61.7	56.1	62.7	60.5

Table 5. Maximum likelihood phylogenetic reconstruction analysis of filtered MSA by different TCS threshold on 853 yeast set. RF: average Robinson-Foulds distance respect to yeast ToL. TPs: the number of genes whose tree topology is identical with yeast subToL (cf Supplemental Table 5 for complete table with other measurements besides RF and TP).

	ClustalW		MAFFT		Muscle		PRANK		SATe		Average	
cutoff	RF	TP	RF	TP	RF	TP	RF	TP	RF	TP	RF	TP
1	0.893	644	0.800	666	0.938	642	0.792	665	0.856	660	0.856	655.4
2	**0.893**	**646**	0.785	672	0.921	646	**0.785**	**666**	**0.785**	**670**	**0.834**	**660.0**
3	0.921	642	0.769	**673**	0.912	**650**	0.830	659	0.837	662	0.854	657.2
4	0.926	640	**0.739**	672	**0.905**	648	0.858	647	0.837	661	0.853	653.6
5	0.947	640	0.816	659	0.950	642	0.891	642	0.863	655	0.893	647.6
6	0.910	646	0.882	650	0.985	635	0.938	634	0.896	643	0.922	641.6
7	0.973	631	0.853	643	0.957	636	0.954	630	0.858	642	0.919	636.4
8	1.116	604	0.957	628	1.036	613	1.046	614	0.957	624	1.022	616.6
9	1.168	581	1.128	588	1.125	587	1.140	589	1.072	597	1.126	588.4

Table 6. Maximum likelihood phylogenetic reconstruction analysis of different post-processing methods on 853 yeast set. RF: average Robinson-Foulds distance with respect to yeast ToL (range 0~8). TPs: the number of genes whose tree topology is identical with yeast subToL (cf Supplemental Table 6 for other detailed metrics). O/O: RF of post-processing is significantly better than original RF, X/X: original RF is significantly better than RF of post-processing (Sign test in 0.1/0.05 significance level, by R *binom.test* function: alternative = "greater"). Best value in Filtered or Weighted are in bold, best value in both Filtered and Weighted are in bold italic.

	ClustalW		MAFFT		Muscle		PRANK		SATe		Average	
	RF	TP	RF	TP	RF	TP	RF	TP	RF	TP	RF	TP
Original	0.900	643	0.797	665	0.952	639	0.792	665	0.858	660	0.860	654.4
Filtered												
HoT	1.002^{xx}	625	0.964^{xx}	627	-	-	-	-	-	-	0.983^{xx}	626.0
Guidance	0.975^{x}	631	0.957^{xx}	625	1.011^{xx}	618	0.920^{xx}	639	-	-	0.966^{xx}	628.3
Gblocks Relaxed	0.994^{xx}	629	0.835	653	0.914	646	0.882^{xx}	642	0.872	650	0.899^{xx}	644.0
Gblocks stringent	1.242^{xx}	584	1.256^{xx}	573	1.256 ^{xx}	578	1.277^{xx}	565	1.284^{xx}	578	1.263^{xx}	575.6
trimAl gappyout	0.954^{xx}	628	0.832	657	0.964	633	0.839^{xx}	648	0.849	655	0.888^{xx}	644.2
trimAl strictplus	1.308^{xx}	561	1.283^{xx}	562	1.294 ^{xx}	559	1.191^{xx}	575	1.247^{xx}	567	1.265^{xx}	564.8
TCS (cutoff=2)	***0.893***	**646**	***0.785***^{oo}	***672***	**0.921**^{o}	**646**	***0.785***	***666***	***0.785***^{oo}	***670***	**0.834**^{oo}	**660.0**
Weighted												
HoT	0.973^{xx}	633	0.933^{x}	634	-	-	-	-	-	-	0.953^{x}	633.5
Guidance	0.947	642	0.947^{x}	628	0.933	641	0.812	656	-	-	0.910^{x}	641.8
trimAl gappyout	**0.896**	644	0.781	669	0.945	642	**0.788**	***666***	0.851	661	0.852	656.4
TCS	0.917	***649***	**0.762**	**670**	***0.842***^{oo}	***664***	0.804	664	***0.785***	**668**	***0.822***^{o}	***663.0***

Figures

Fig. 1. Specificity and Sensitivity of the TCS indexes with respect to structural correctness, as measured on ClustalW, MAFFT, MUSCLE, PRANK and SATe MSAs. Specificity and Sensitivity are represented as square and circle, respectively. All points correspond to measurements made by removing all residues within the target MSA having a ResidueTCS(*10) score lower than the considered threshold.

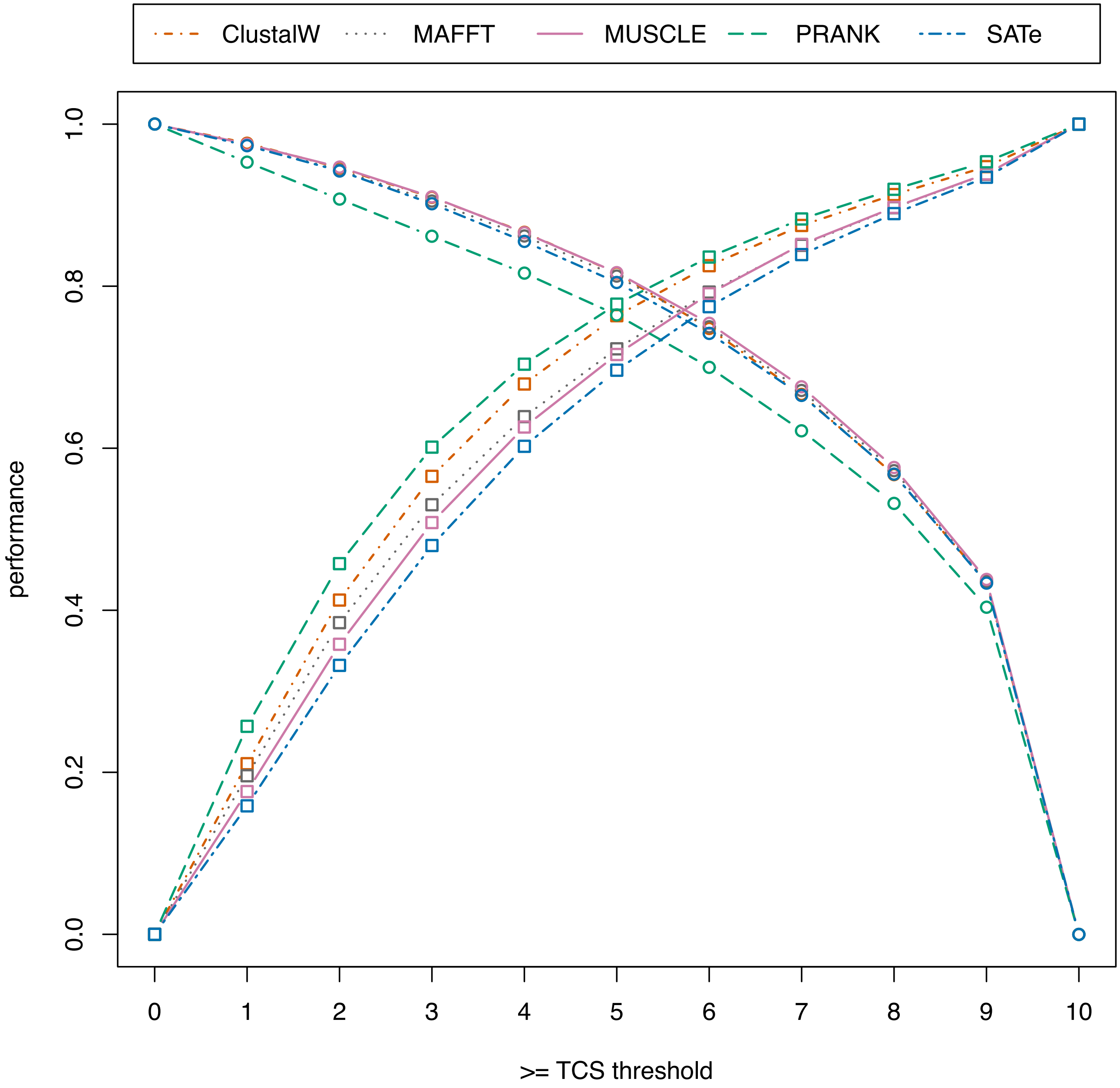


Fig. 2. Comparison between Δ SPS and Δ confidences by (a) GUIDANCE and (b) TCS on BAliBASE 3 using alignments produced by MAFFT, MUSCLE and ClustalW as well as the reference alignment. Each point represents one comparison of two alternative MSAs. All points in the top right and bottom left quadrant (same algebraic sign) correspond to pairs of datasets for which the relative TCS score and the relative accuracy scores are in agreement.

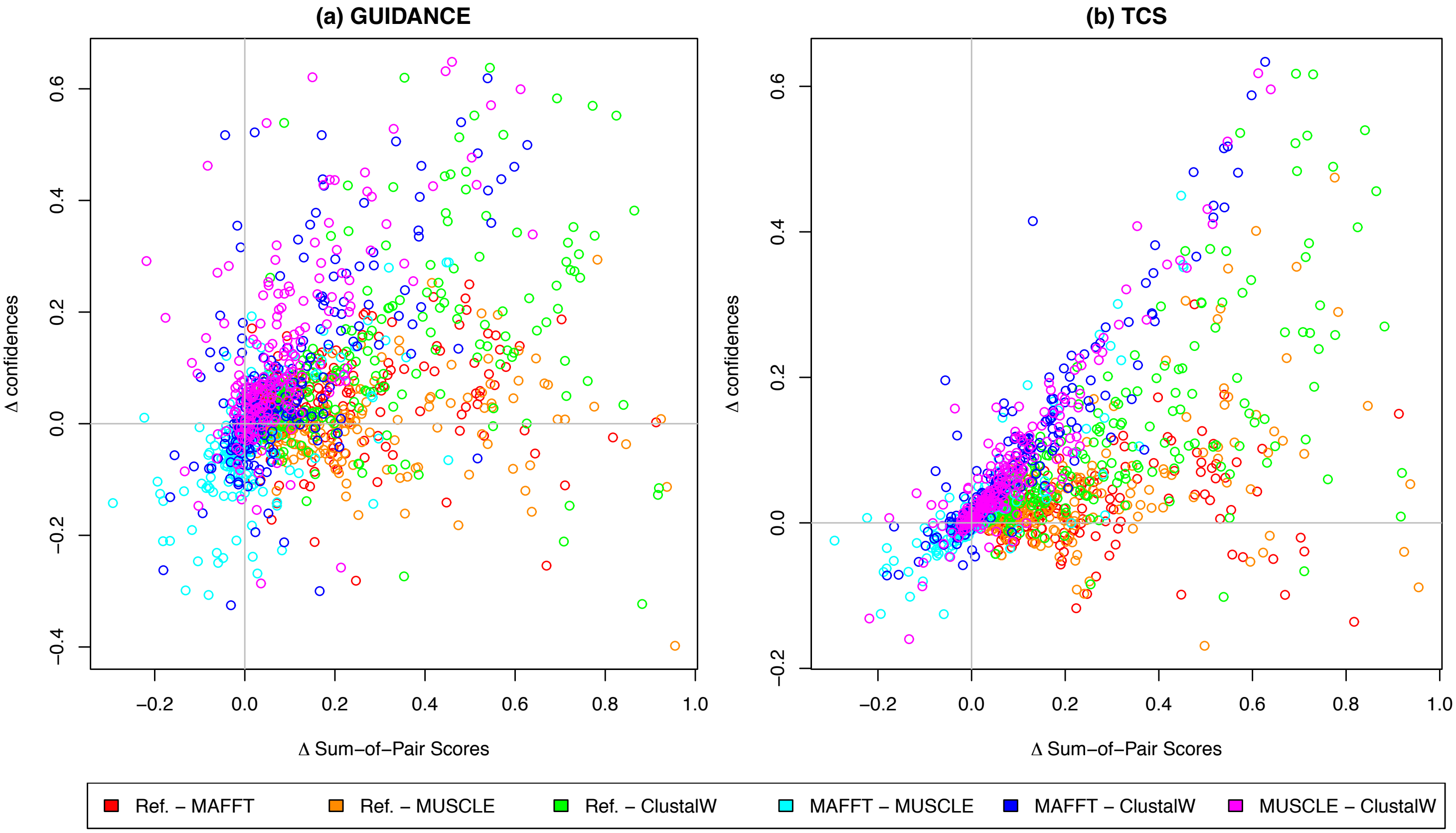


Fig. 3. Average normalized Robinson-Foulds distance to reference tree with 16, 32 and 64 tips from the Maximum Likelihood trees calculated with the MAFFT complete alignments, the same alignments after treatments with different methods. The asymmetric trees with three different divergence levels (0.5, 1.0 and 2.0) were used for the simulations with different alignment lengths (400, 800 and 1200). The performance of the tool in filtering and weighting schemes are plotted as the same color dot and dash lines, respectively. OA: original alignment, DF: GUIDANCE filtering, HF: HoT filtering, GR: Gblocks relaxed, GS: Gblocks stringent, TG: trimAl gappyout, TS: trimAl strictplus, W2: TCS filtering when cutoff equaling 2, DW: GUIDANCE weighting, HW: HoT weighting, TW: trimAl weighting, WR: TCS weighting.

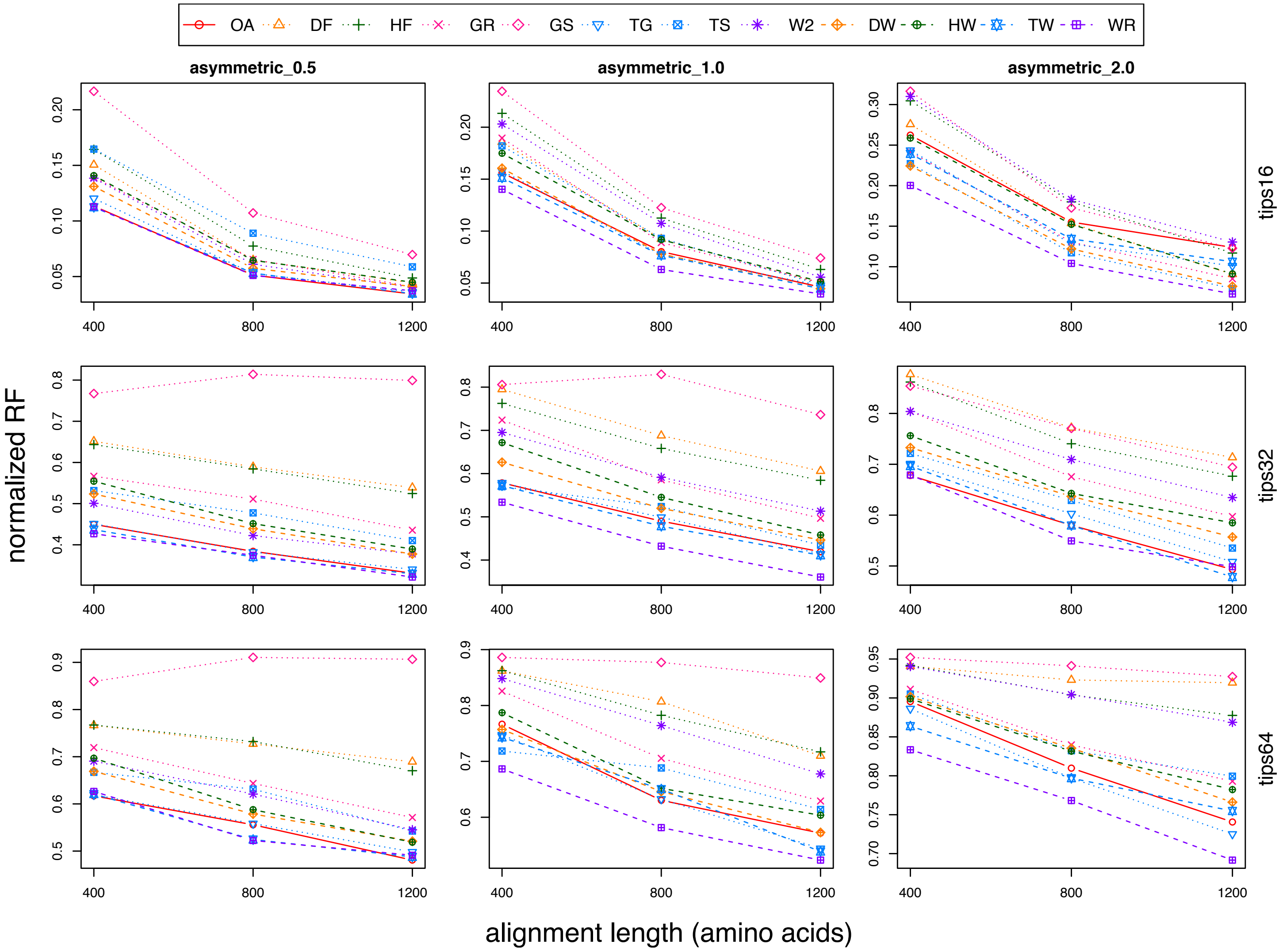
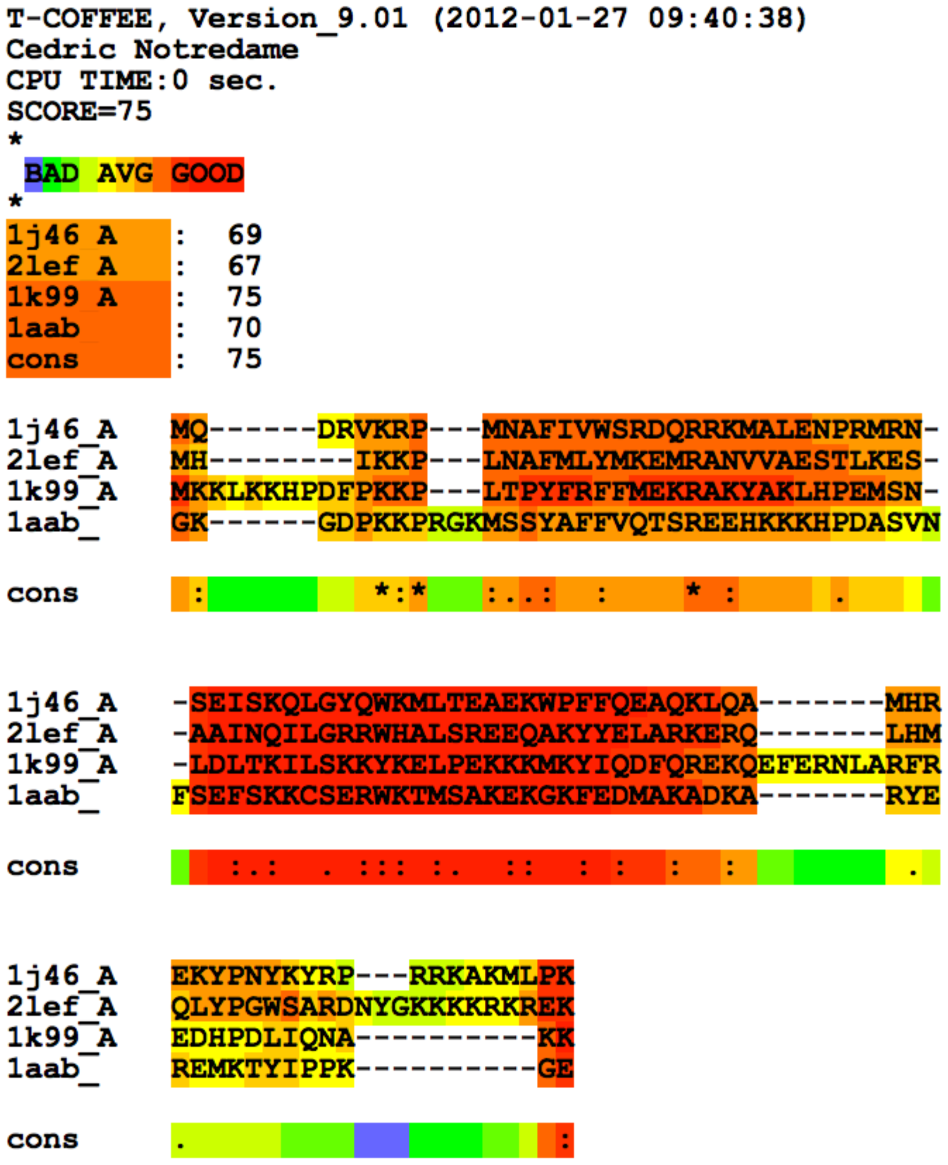


Fig. 4. The TCS color code in the HTML output (.score_thml) indicates the agreement between the library and the considered alignment (MAFFT alignment on BAliBASE, BB11001). The TCS scale goes from 0 (blue) to 9 (red). It must be stressed that this score depends both on the MSA and on the library.



```
T-COFFEE, Version_9.01 (2012-01-27 09:40:38)
Cedric Notredame
CPU TIME:0 sec.
SCORE=75
*
 BAD AVG GOOD
*
1j46 A    :  69
2lef A    :  67
1k99 A    :  75
1aab      :  70
cons      :  75

1j46_A    MQ------DRVKRP---MNAFIVWSRDQRRKMALENPRMRN-
2lef_A    MH--------IKKP---LNAFMLYMKEMRANVVAESTLKES-
1k99_A    MKKLKKHPDFPKKP---LTPYFRFFMEKRAKYAKLHPEMSN-
1aab_     GK------GDPKKPRGKMSSYAFFVQTSREEHKKKHPDASVN

cons       :         *:*   :..:  :    * :     .


1j46_A    -SEISKQLGYQWKMLTEAEKWPFFQEAQKLQA-------MHR
2lef_A    -AAINQILGRRWHALSREEQAKYYELARKERQ-------LHM
1k99_A    -LDLTKILSKKYKELPEKKKMKYIQDFQREKQEFERNLARFR
1aab_     FSEFSKKCSERWKTMSAKEKGKFEDMAKADKA-------RYE

cons         :.:  . ::: :.  ::  : :  :  :         .


1j46_A    EKYPNYKYRP---RRKAKMLPK
2lef_A    QLYPGWSARDNYGKKKKRKREK
1k99_A    EDHPDLIQNA----------KK
1aab_     REMKTYIPPK----------GE

cons      .                    :
```