Published online 22 April 2010 *Nucleic Acids Research, 2010, Vol. 38, No. 15* ***4917–4928***
doi:10.1093/nar/gkq255

PicXAA: greedy probabilistic construction of maximum expected accuracy alignment of multiple sequences

Sayed Mohammad Ebrahim Sahraeian and Byung-Jun Yoon*

Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX 77843, USA

Received December 23, 2009; Revised March 25, 2010; Accepted March 26, 2010

ABSTRACT

Accurate tools for multiple sequence alignment (MSA) are essential for comparative studies of the function and structure of biological sequences. However, it is very challenging to develop a computationally efficient algorithm that can consistently predict accurate alignments for various types of sequence sets. In this article, we introduce PicXAA (Probabilistic Maximum Accuracy Alignment), a probabilistic non-progressive alignment algorithm that aims to find protein alignments with maximum expected accuracy. PicXAA greedily builds up the multiple alignment from sequence regions with high local similarities, thereby yielding an accurate global alignment that effectively grasps the local similarities among sequences. Evaluations on several widely used benchmark sets show that PicXAA constantly yields accurate alignment results on a wide range of reference sets, with especially remarkable improvements over other leading algorithms on sequence sets with local similarities. PicXAA source code is freely available at: http://www.ece.tamu.edu/~bjyoon/picxaa/.

INTRODUCTION

Multiple sequence alignment (MSA) has become an essential tool in various areas of molecular biology research, including the reconstruction of phylogenetic trees, predicting the structure of biomolecules, detection of key functional regions, identification of conserved sequence motifs and homology modeling (1–8). The main goal of an MSA algorithm can be viewed as detecting and aligning the homologous regions across different sequences, which have equivalent structures and/or similar functional roles. This is typically achieved by optimizing an objective function that measures the quality of the alignment, either explicitly or implicitly. One of the primary challenges in sequence alignment is to find a biologically meaningful objective function. A common choice of many alignment algorithms has been the 'sum-of-pairs' (SP) score, which simply takes the sum of the scores of all pairwise alignments in a given multiple alignment. The optimal alignment that maximizes the SP score can be found using dynamic programming (9,10). However, this problem is NP-complete (11) and the dynamic programming approach becomes quickly infeasible once the number of sequences increases. For this reason, several heuristic techniques have been proposed as possible alternatives, which aim to find a good approximate solution at a reasonable computational cost (12–17). One of the most popular techniques that have been used to reduce the overall complexity is the progressive alignment scheme (16,17), which has been adopted by various alignment algorithms such as CLUSTALW (18), T-Coffee (19), ProbCons (20), MUSCLE (21), MAFFT (22–24), MUMMALS (25) and Pecan (26), just to name a few. The basic idea of the progressive scheme is to build a guide tree based on the similarities among sequences and to grow the MSA by repetitively aligning pairs of sequences or sequence profiles along the tree. Despite its usefulness, one significant weakness of the progressive alignment approach is that it tends to propagate the errors made in the early stages throughout the entire process, which may significantly degrade the quality of the final alignment.

A number of strategies have been proposed to overcome this problem, where the iterative refinement technique and the consistency-based approach have been shown to be especially useful. The 'iterative refinement' technique is carried out as a postprocessing step (20–22,25), during which it repetitively divides the aligned sequences into two random groups and realigns them. Unlike the iterative refinement approach, the 'consistency-based' approach tries to reduce the chance of early errors when constructing the alignment (19,20,25–28), instead of correcting existing errors via postprocessing. This is typically achieved by updating the pairwise sequence comparison scores based on other sequences in the alignment, to obtain pairwise alignments that are consistent with one another. T-Coffee (19) is one of the first methods that

*To whom correspondence should be addressed. Tel: +1 979-845-6942; Fax: +1 979-845-6259; Email: bjyoon@ece.tamu.edu

implemented this idea by using a consistency-based alignment measure based on a library of pairwise alignments. With a similar motivation, ProbCons (20) introduced the probabilistic consistency transformation. This algorithm transforms the pairwise residue alignment probabilities, computed using pair hidden Markov models (pair-HMMs), by probabilistically incorporating the information from other sequences. MUMMALS (25) adopts the same approach as ProbCons, but it computes the alignment probabilities using more complex HMMs that also consider local secondary structure similarities. ProbAlign (27) also uses the probabilistic consistency transformation scheme to construct MSAs, where a partition function-based methodology is used to estimate the residue alignment probabilities. In addition to the iterative refinement and consistency-based techniques, DIALIGN (29–31) proposed another solution to overcome the shortcomings of the progressive alignment approach, which constructs the global alignment by assembling the local segments with high similarity. Recently AMAP (32) introduced another non-progressive alignment technique, called sequence annealing, which incrementally constructs the multiple sequence alignment by merging single columns. This technique was adopted and further developed in FSA (33). It has been shown that such incremental approach can effectively reduce the number of incorrect residue alignments (32,33).

As mentioned earlier, MSA algorithms generally aim to find the optimal alignment by optimizing an objective function. Optimality of the alignment, however, can be defined in different ways. For example, we may define the optimal alignment as the alignment with the highest probability or the one with the largest expected number of correctly aligned residues. According to the first definition, our goal would be to develop an algorithm that can find the MSA that has the maximum probability of being identical to the true (unknown) alignment among all possible alignments. In practice, it is not possible to pinpoint the correct alignment with certainty, in which case, it would be more useful to find an alignment that shares as many similarities as possible with the true alignment. In other words, it would be more beneficial to find the alignment that maximizes the expected accuracy, or the expected number of correctly aligned residue pairs. Do *et al.* (20) implemented this idea in ProbCons to construct the so-called 'maximum expected accuracy' (MEA) alignment, using consistency-transformed pairwise residue alignment probabilities. The MEA criterion has been also adopted in many other MSA algorithms, such as ProbAlign (27), Pecan (26), and NRAlign (34), which demonstrated that the MEA-based approach can improve the average accuracy of the predicted MSA. Note that all the aforementioned algorithms that try to construct MEA alignments are mainly based on the progressive technique (20,26,27,34). The *minimum expected distance* criterion adopted by the sequence annealing technique is essentially identical to the MEA criterion, hence AMAP (32) and FSA (33) can be viewed as examples of non-progressive algorithms for building MEA alignments.

In this article, we introduce PicXAA (**p**robabilist**ic** ma**x**imum **a**ccuracy **a**lignment), an effective non-progressive algorithm that aims to find MEA alignment of multiple sequences. PicXAA greedily builds up the alignment from sequence regions with high local similarities, thereby yielding an accurate global alignment that effectively grasps local similarities among sequences. For a fast and accurate alignment, we adopt an efficient graph-based construction scheme, and also introduce a novel probabilistic consistency transformation and a robust refinement technique. To demonstrate the effectiveness of the proposed techniques, we conduct extensive experiments to compare PicXAA with other well-known alignment algorithms based on several alignment benchmarks. Experimental results confirm that PicXAA consistently yields accurate alignment results on various reference sets, with significant improvements in sequence sets with local similarities.

METHODS

The main goal of PicXAA is to find the MSA with the MEA, which maximizes the expected number of correctly aligned residue pairs. To effectively capture local similarities in the final alignment, while avoiding the propagation of early-stage errors that is often observed in progressive algorithms, we take a greedy approach to probabilistically build up the MSA, by starting from confidently alignable regions (with high similarities) and proceeding toward less confident regions (with lower similarities). The following subsections provide a brief overview of the proposed algorithm.

Improved probabilistic consistency transformation

To align m sequences in a set $\mathbf{S} = \{s_1, \ldots, s_m\}$, we first need to obtain the posterior pairwise alignment probability matrix $P_{\mathbf{xy}}$ for each sequence pair $(\mathbf{x},\mathbf{y})$ for all $\mathbf{x},\mathbf{y} \in \mathbf{S}$. $P_{\mathbf{xy}}(i,j) = P(x_i \sim y_j \in a^* | \mathbf{x},\mathbf{y})$ is the probability that residues $x_i \in \mathbf{x}$ and $y_j \in \mathbf{y}$ are matched in the true (unknown) alignment a^*. For simplicity, we hereafter denote the alignment probability as $P(x_i \sim y_j)$ instead of $P(x_i \sim y_j \in a^*)$. These posterior alignment probabilities can be estimated using various methods. Further discussion on several possible methods to compute these probabilities can be found in Supplementary Data.

Given the pairwise residue alignment probabilities, the probabilistic consistency transformation modifies the probabilities using the information from other sequences to make them suitable for constructing a more consistent and accurate MSA. One important observation in the consistency-based alignment approach is that all the pairwise alignments induced from a given MSA should be consistent with each other. This means that if residue x_i ($\in \mathbf{x}$) aligns with residue z_k ($\in \mathbf{z}$) in the $\mathbf{x}-\mathbf{z}$ alignment, and if z_k aligns with residue y_j ($\in \mathbf{y}$) in the $\mathbf{z}-\mathbf{y}$ alignment, then x_i must align with y_j in the $\mathbf{x}-\mathbf{y}$ alignment. We can thus utilize the 'intermediate' sequence $\mathbf{z}$ to improve the $\mathbf{x}-\mathbf{y}$ alignment by making it consistent with the alignments $\mathbf{x}-\mathbf{z}$ and $\mathbf{z}-\mathbf{y}$.

Based on this motivation, Do *et al.* (20) introduced the so called probabilistic consistency transformation that modifies the alignment probability for a residue pair

$x_i \sim y_j$, by incorporating the alignment probability between x_i and z_k and that between z_k and y_j. This transformation can be written as:

$$P(x_i \sim y_j|\mathbf{x},\mathbf{y},\mathbf{z}) = \sum_{z_k} P(x_i \sim z_k|\mathbf{x},\mathbf{z})P(z_k \sim y_j|\mathbf{z},\mathbf{y}).$$

For multiple intermediate sequences, the overall probabilistic consistency transformation is defined as (20):

$$P'(x_i \sim y_j|S) = \frac{1}{|\mathbf{S}|}\sum_{\mathbf{z}\in\mathbf{S}} P(x_i \sim y_j|\mathbf{x},\mathbf{y},\mathbf{z}),$$

where $\mathbf{S}$ is the set of all sequences. This transformation assumes that every intermediate sequence $\mathbf{z}$ is homologous to both $\mathbf{x}$ and $\mathbf{y}$, hence it contains useful homology information that can be used to obtain an accurate and consistent $\mathbf{x}-\mathbf{y}$ alignment. However, when $\mathbf{S}$ consists of distantly related sequences, this assumption does not necessarily hold, and the transformation can even degrade the quality of the resulting MSA. For instance, if sequences in two distantly related subfamilies are to be aligned, the alignment probability between sequences in the same family can be significantly degraded if we incorporate sequences in the other family. To address this problem, we propose a new probabilistic consistency transformation that explicitly considers the relative significance of each intermediate sequence $\mathbf{z}$ in improving the $\mathbf{x}-\mathbf{y}$ alignment.

Let $\mathbf{Z} = \{\mathbf{z}\in\mathbf{S}|\mathbf{x}\diamondsuit\mathbf{z} \wedge \mathbf{y}\diamondsuit\mathbf{z}\}$ be the set of sequences in $\mathbf{S}$ that are related to both $\mathbf{x}$ and $\mathbf{y}$, where $\mathbf{x}\diamondsuit\mathbf{z}$ means $\mathbf{x}$ and $\mathbf{z}$ are homologous. Using only the homologous sequences, in the set $\mathbf{Z}$, we define the probabilistic consistency transformation as:

$$P'(x_i \sim y_j|\mathbf{S}) = \frac{1}{|\mathbf{Z}|}\sum_{\mathbf{z}\in\mathbf{Z}} P(x_i \sim y_j|\mathbf{x},\mathbf{y},\mathbf{z}).$$

This transformation can be also written as:

$$P'(x_i \sim y_j|\mathbf{S}) = \frac{\sum_{\mathbf{z}\in\mathbf{S}} P(x_i \sim y_j|\mathbf{x},\mathbf{y},\mathbf{z})\mathbf{I}\{\mathbf{x}\diamondsuit\mathbf{z} \wedge \mathbf{y}\diamondsuit\mathbf{z}\}}{\sum_{\mathbf{z}\in\mathbf{S}} \mathbf{I}\{\mathbf{x}\diamondsuit\mathbf{z} \wedge \mathbf{y}\diamondsuit\mathbf{z}\}}, \quad (1)$$

using the identity function $\mathbf{I}\{\cdot\}$, where $\mathbf{I}\{\mathbf{x}\diamondsuit\mathbf{z} \wedge \mathbf{y}\diamondsuit\mathbf{z}\} = 1$ if $\mathbf{z}$ is homologous to both $\mathbf{x}$ and $\mathbf{y}$, and $\mathbf{I}\{\mathbf{x}\diamondsuit\mathbf{z} \wedge \mathbf{y}\diamondsuit\mathbf{z}\} = 0$ otherwise. In practice, we cannot judge with certainty whether two sequences are homologous or not. For this reason, it is useful to describe the relationship between the sequences probabilistically, instead of directly using the binary indicator function. Therefore, we use the expectation $\mathbf{E}[\mathbf{I}\{\mathbf{x}\diamondsuit\mathbf{z}\}] = P(\mathbf{x}\diamondsuit\mathbf{z})$ as a practical alternative, where $P(\mathbf{x}\diamondsuit\mathbf{z})$ is the probability that $\mathbf{x}$ and $\mathbf{z}$ are homologous to each other. We estimate this probability $P(\mathbf{x}\diamondsuit\mathbf{z})$ based on the similarity between the sequences $\mathbf{x}$ and $\mathbf{z}$ as:

$$P(\mathbf{x}\diamondsuit\mathbf{z}) \triangleq \frac{1}{|\bar{a}|}\sum_{x_i\sim z_k\in\bar{a}} P(x_i \sim z_k|\mathbf{x},\mathbf{z}),$$

where $\bar{a}$ is the optimal pairwise alignment of $\mathbf{x}$ and $\mathbf{z}$. Note that, when estimating $P(\mathbf{x}\diamondsuit\mathbf{z})$, we only consider the residue pairs that are aligned in the predicted optimal alignment $\bar{a}$. By replacing the identity functions with their expected values in Equation (1) and by assuming the independence between $\mathbf{x}\diamondsuit\mathbf{z}$ and $\mathbf{y}\diamondsuit\mathbf{z}$, we get:

$$P'(x_i \sim y_j|\mathbf{S}) \simeq \frac{\sum_{\mathbf{z}\in\mathbf{S}} P(x_i \sim y_j|\mathbf{x},\mathbf{y},\mathbf{z})P(\mathbf{x}\diamondsuit\mathbf{z})P(\mathbf{y}\diamondsuit\mathbf{z})}{\sum_{\mathbf{z}\in\mathbf{S}} P(\mathbf{x}\diamondsuit\mathbf{z})P(\mathbf{y}\diamondsuit\mathbf{z})}.$$

Therefore, by using $P(x_i \sim y_j|\mathbf{x},\mathbf{y},\mathbf{z}) = \sum_{z_k} P(x_i \sim z_k|\mathbf{x},\mathbf{z})P(z_k \sim y_j|\mathbf{z},\mathbf{y})$, we obtain the following probabilistic consistency transformation:

$$P'(x_i \sim y_j|\mathbf{S}) \simeq \frac{\sum_{\mathbf{z}\in\mathbf{S}}\sum_{z_k} P(x_i \sim z_k|\mathbf{x},\mathbf{z})P(z_k \sim y_j|\mathbf{z},\mathbf{y})P(\mathbf{x}\diamondsuit\mathbf{z})P(\mathbf{y}\diamondsuit\mathbf{z})}{\sum_{\mathbf{z}\in\mathbf{S}} P(\mathbf{x}\diamondsuit\mathbf{z})P(\mathbf{y}\diamondsuit\mathbf{z})}.$$

Conceptually, the new probabilistic consistency transformation improves the consistency of the alignment $\mathbf{x}-\mathbf{y}$ with other pairwise alignments in the MSA, by transforming the residue alignment probabilities between $x_i \in \mathbf{x}$ and $y_j \in \mathbf{y}$ using only the information from sequences that are homologous to both $\mathbf{x}$ and $\mathbf{y}$. In this way, we can obtain more probabilistically consistent estimate of the posterior alignment probabilities, which ultimately helps enhance the quality of the final MSA. As in (20), we can efficiently implement this transformation based on sparse matrix multiplication, since most values in the matrices $P_{\mathbf{x},\mathbf{z}}$ and $P_{\mathbf{z},\mathbf{y}}$ will be close to zero. The transformation has a computational complexity of $O(\mu^2 Lm^3)$, where μ is the average number of non-zero elements per row (typically $1 \leq \mu \leq 5$ in real examples), m is the number of sequences, and L is the length of each sequence.

More detailed derivation of the improved probabilistic consistency transformation and further discussion can be found in Supplementary Data.

Construction of the alignment graph

Given a set of sequences $\mathbf{S}$, our ultimate goal is to find the MSA that maximizes the expected accuracy $\mathbf{E}[\text{accuracy}(a,a^*)|\mathbf{S}]$ (i.e. the expected number of correctly aligned residues) over all sequences in $\mathbf{S}$. To this aim, we construct the alignment by successively adding the most confident pairwise residue alignments. This greedy alignment approach is conceptually similar to the one used in sequence annealing (32,33). However, unlike sequence annealing, which greedily merges pairs of columns, we always add a *single residue pair* at a time, based on the consistency-transformed posterior alignment probabilities. During this process, we may encounter residue pairs that are incompatible with the current alignment. Verifying the compatibility of a new residue pair with the current alignment can be computationally expensive. For fast compatibility verification and efficient construction of the MSA, we adopt a graph-based framework described in the following. Note that a similar approach was also used in sequence annealing (32,33).

Let us consider all possible residue pairs (x_i,y_j) for all $\mathbf{x},\mathbf{y} \in \mathbf{S}$ and all $i \in \{1,\ldots,|\mathbf{x}|\}$, $j \in \{1,\ldots,|\mathbf{y}|\}$. First, we sort these pairs according to their consistency-transformed alignment probability $P'_{\mathbf{xy}}(i,j)$ to obtain an ordered set

$\mathbf{P} = \{p_1, p_2, \ldots, p_n\}$, where p_1 corresponds to the most probable residue pair and p_n corresponds to the least probable pair. To construct the MSA, we start by inserting the most probable residue pair p_1 in the alignment. Then we examine the next residue pair p_2 and add it to the alignment only if it is compatible with the current alignment. Otherwise, we discard p_2 and move on to the next most probable residue pair in $\mathbf{P}$. We continue this process for every residue pair $p_i \in \mathbf{P}$. The multiple sequence alignment is modeled as a set $\mathbf{C} = \{c^{(1)}, c^{(2)}, \ldots\}$ of aligned 'residue groups'. Each group $c^{(i)}$ consists of residues from different sequences that are aligned to each other, hence will be placed in the same column in the final MSA. We let $c^{(i)} = \{r_{j_1}^{(k_1)}, r_{j_2}^{(k_2)}, \ldots\}$, where $r_j^{(k)}$ is the j^{th} residue of the sequence $\mathbf{s}_k$. For convenience, we refer to each residue group $c^{(i)}$ as a column. MSA is represented as a directed acyclic graph $\mathcal{G}$, whose nodes correspond to the columns $c^{(i)}$ in the current alignment. For convenience, we use the terms 'node' and 'column' interchangeably. The edges are defined such that $c^{(i)} \rightarrow c^{(j)}$ implies that column $c^{(i)}$ precedes column $c^{(j)}$ in the given alignment. It can be easily shown that incompatibilities in the MSA introduce cycles in the alignment graph $\mathcal{G}$. Therefore, we can maintain the compatibility of the MSA by adding only the residue pairs that keep the graph $\mathcal{G}$ acyclic.

Adding a new candidate residue pair $p^* = (a,b)$ to the current alignment will lead to one of the following four possibilities:

(1) *Adding a new column:* If neither residue a nor residue b is included in the current alignment ($\nexists i : a \in c^{(i)} \vee b \in c^{(i)}$), the residue pair $p^* = (a,b)$ should be added to the alignment as a new column, provided that it is compatible with the current MSA. Let us denote the new node for this column as $c^* = \{a,b\}$ and assume that residues a and b belong to sequences $\mathbf{s}_a$ and $\mathbf{s}_b$, respectively. To find the incoming and outgoing edges for this new node, we should identify the columns in the current alignment that contain residues in $\mathbf{s}_a$ or $\mathbf{s}_b$. This allows us to determine the relative position of the new column with respect to other existing columns. To determine the relative position of c^*, we only need to identify the closest preceding and succeeding columns that contain residues in $\mathbf{s}_a$ and $\mathbf{s}_b$. Let c_l^a be the column that contains the closest residue in $\mathbf{s}_a$ that precedes a, and c_r^a be the column that contains the closest residue in $\mathbf{s}_a$ that comes after a. Similarly, we define c_l^b and c_r^b as the closest preceding and succeeding columns from the residue b, respectively. Now we can figure out the relative position of the new column c^* based on the four columns c_l^a, c_r^a, c_l^b and c_r^b. Then we simultaneously consider the four new edges, $c_l^a \rightarrow c^*$, $c^* \rightarrow c_r^a$, $c_l^b \rightarrow c^*$ and $c^* \rightarrow c_r^b$, to check whether these edges lead to any cycles. If the addition of the new column c^* introduces a cycle in $\mathcal{G}$, we discard the residue pair p^* since it is not compatible with the current alignment, and we move on to the next residue pair in $\mathbf{P}$. Otherwise, the new node c^* is retained in the graph. Figure 1A illustrates this process.

In order to reduce the computational cost for verifying the compatibility, we prefer $\mathcal{G}$ to have as few multipaths as possible between any two nodes in the graph. For this purpose, we prune the alignment graph after adding a new node (details described in Supplementary Data). During pruning, we try to remove redundant edges, which do not add any information about the relative position among the columns in the alignment graph $\mathcal{G}$.

(2) *Extending an existing column*: suppose only one of the residues in $p^* = (a,b)$ appears in the current alignment ($(\exists i : a \in c^{(i)}; \nexists j : b \in c^{(j)})$ or $(\exists i : b \in c^{(i)}; \nexists j : a \in c^{(j)})$). Without loss of generality, we assume that only a is included in the alignment. Let us denote the column that contains a as $c^{(i)}$. In this case, the other residue b should be added to the same column as a, hence we need to extend $c^{(i)}$ to include b. Extension of the column may add new edges to the node $c^{(i)}$, and we have to examine whether this introduces any cycles in the graph $\mathcal{G}$. Again, suppose a and b belong to sequences $\mathbf{s}_a$ and $\mathbf{s}_b$, respectively. As before, we look for the closest preceding and succeeding columns (with respect to b) that contain residues in $\mathbf{s}_b$. We denote these columns as c_l^b and c_r^b. If the new edges $c_l^b \rightarrow c^{(i)}$ and $c^{(i)} \rightarrow c_r^b$ do not introduce any cycle in the graph, we keep b in column $c^{(i)}$. Otherwise, we discard the residue pair p^*. Figure 1B illustrates this process. After extending the column, we again prune the graph $\mathcal{G}$ as described in Supplementary Data.

(3) *Merging two columns*: suppose the residues a and b are included in two distinct columns c^a and c^b,

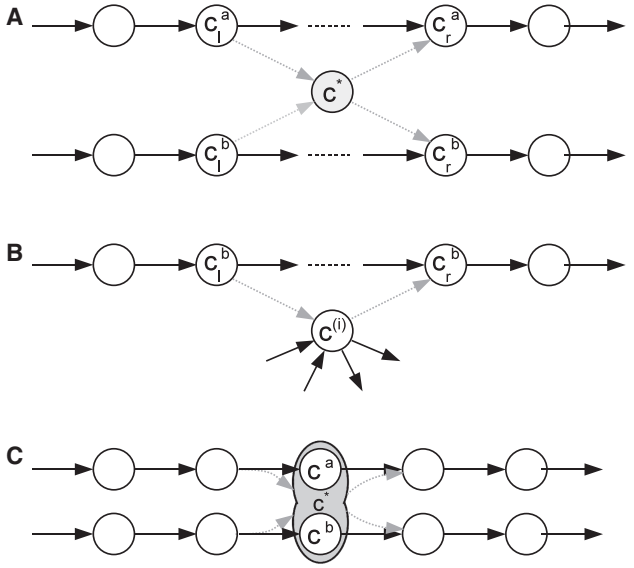

Figure 1. Graph Construction Process. (**A**) Adding a new column (node) c^* and the corresponding edges to the alignment graph. (**B**) Extending the column (node) $c^{(i)}$ which may introduce new edges. (**C**) Merging the columns (nodes) c^a and c^b into a single column (node) c^*.

respectively ($\exists i,j : i \neq j, a \in c^{(i)}, b \in c^{(j)}$). In order to add the residue pair $p^* = (a,b)$ to the current alignment in a compatible manner, we have to merge the columns c^a and c^b into a single column. This corresponds to merging the corresponding nodes c^a and c^b in the alignment graph $\mathcal{G}$ into a new node c^*. In order to verify whether merging the columns leads to a legitimate MSA, we examine whether adding an edge between c^a and c^b introduces any cycles. If the graph remains acyclic, we align all the residues in c^a and c^b in the same column denoted as c^*. Otherwise, we discard the residue pair p^*. Figure 1C illustrates the merging process. After merging the columns, the alignment graph is pruned as described in Supplementary Data.

(4) *Residue pair p^* already included in the current alignment*: no action is needed in this case, since the pair (a,b) is already included in the current alignment ($\exists i : a,b \in c^{(i)}$).

Mapping the alignment graph to an MSA

Once the graph construction process is complete, the resulting alignment graph $\mathcal{G}$ can serve as a skeleton of the final MSA. Each node in $\mathcal{G}$ contains the residues that should be aligned in the same column in the final MSA. Suppose there is a sequence, none of whose residues is included in a given column. This implies that a gap symbol should be placed in the given column for this sequence. To obtain the final MSA from the graph $\mathcal{G}$, we should arrange the nodes in a linear directed path $\mathcal{P}$ according to a legitimate topological ordering. More precisely, we should find a path $\mathcal{P} : c^{(k_1)} \rightarrow c^{(k_2)} \rightarrow c^{(k_3)} \rightarrow \cdots$, so that there is no path from $c^{(k_j)}$ to $c^{(k_i)}$ in $\mathcal{G}$ for any $i < j$. This guarantees that the order of nodes in $\mathcal{P}$ do not contradict their relative order in $\mathcal{G}$. We can easily find such $\mathcal{P}$ using the depth-first search algorithm starting from one of the root nodes in $\mathcal{G}$. Theoretically, we may have residues that are not included in any node due to their small alignment probability to other residues. These residues can be inserted in $\mathcal{P}$ as single residue columns, according to their relative position to other residues. The resulting linear path $\mathcal{P}$ uniquely determines the final MSA. An illustrative example for the graph construction process using PicXAA can be found in the Supplementary Data (Supplementary Section S3 and Figure S4).

Improving the alignment quality in low confidence regions

The proposed greedy graph construction scheme is very effective in capturing local similarities among sequences and it faithfully preserves the confidently alignable residue pairs in the MSA. However, residue pairs with low alignment probabilities may have many other competing residue pairs, and the greedy selection of the most probable residue pair among these competing pairs may not necessarily yield an accurate alignment result. This can be a problem for sequence sets that consist of distant sequence families, since sequences that belong to different families will typically have low alignment probabilities. For such datasets, we can improve the alignment accuracy by first grouping similar sequences and performing a profile–profile alignment between distinct groups. This will allow us to improve the alignment quality in low confidence regions (between distantly related sequences that belong to different families), while preserving the confidently alignable residue pairs (between closely related sequences in the same family).

Based on this motivation, we propose the following discriminative refinement technique:

(1) For each sequence $\mathbf{x}$ in the current multiple alignment, we find the set of similar sequences using the k-means clustering. We measure the similarity between two sequences by computing the expected accuracy of their pairwise alignment, which is defined as:

$$\mathbf{E}_{a^*}[\mathrm{acc}(a,a^*)|\mathbf{x},\mathbf{y}] = \frac{1}{|a|} \sum_{x_i \sim y_j \in a} P(x_i \sim y_j|\mathbf{x},\mathbf{y}).$$

For each sequence $\mathbf{x} \in \mathbf{S}$, we try to partition the set $\mathbf{S}$ into two clusters based on this similarity measure: $\mathbf{S_x}$, the set of sequences that are similar to $\mathbf{x}$, and $\mathbf{N_x}$ ($= \mathbf{S} - \mathbf{S_x}$), the set of remaining sequences in $\mathbf{S}$ that are not similar to $\mathbf{x}$.

(2) Then we repeat the following steps for all sequences $\mathbf{x} \in \mathbf{S}$:

(i) Realign $\mathbf{x}$ with the current alignment profile of $\mathbf{S_x}$.
(ii) Align the resulting profile of $\mathbf{S'_x}(= \mathbf{S_x} \cup \{\mathbf{x}\})$ with the current alignment profile of $\mathbf{N_x}$.
(iii) Choose another sequence in $\mathbf{S}$ and repeat the previous steps.

The proposed refinement technique, based on a discriminative-split-and-realignment strategy, has a number of advantages over the conventional iterative refinement technique, based on a random-split-and-realignment strategy (20,27). First, it typically converges to an accurate alignment in a few (often in a single) iterations, since the overall process is not randomized. Second, the proposed technique takes advantage of both the intra-family similarity (by realigning $\mathbf{x}$ with $\mathbf{S_x}$) as well as the inter-family similarity (by aligning $\mathbf{S'_x}$ with $\mathbf{N_x}$), thereby improving the alignment quality in low similarity regions without breaking the confidently aligned residues.

RESULTS

Experimental results and comparison

We used six different benchmark datasets: BAliBASE 3.0 (35), IRMBASE 2.0 (31), SABmark 1.65 (36), PREFAB 4.0 (21), HOMSTRAD (37) and OXBench 1.3 (38) to assess the performance of PicXAA in comparison with other well-known MSA algorithms: ProbCons 1.12 (20), ProbAlign 1.1 (27), MAFFT 6.708 (24) with four different options ('-linsi', '-ginsi', '-einsi', and '-fftnsi'), MUMMALS 1.01 (25) with HMM_1_3_1 option, MUSCLE 3.7 (21), T-Coffee 6.00 (19), CLUSTALW

2.0.10 (18) and DIALIGN-TX (31). Note that all of these methods employ the progressive strategy except DIALIGN-TX. DIALIGN-TX, on the other hand, is a segment-based local alignment method that combines a greedy algorithm with the progressive alignment approach.

It is important to note that PicXAA does not depend on a specific method for computing the pairwise residue alignment probabilities, although it can certainly benefit from a more accurate estimation scheme. To demonstrate this point, in this work, we used three different methods for computing the alignment probabilities: (i) the pair-HMM approach implemented in ProbCons (20), (ii) the structural pair-HMM approach in MUMMALS (25), and (iii) the partition function-based scheme used in ProbAlign (27). We refer to these methods as PicXAA-PHMM, PicXAA-SPHMM and PicXAA-PF, respectively. Details on these posterior probability computation schemes can be found in Supplementary Data.

In the following, we compare these three implementations of PicXAA with the aforementioned MSA algorithms. In each table that summarizes the alignment results based on a specific database, the best result is shown in bold and the second best result is shown in underlined italic. For each test set, we chose the best performing version of PicXAA as the reference (marked by asterisk) and compared its performance with other techniques and estimated the statistical significance of the performance difference using the Wilcoxon signed-rank test. In each table, minus symbols are used to denote the statistically significant inferiority of the respective method compared with PicXAA, while plus symbols are used to denote statistically significant superiority of the method. Single plus or minus symbols denote that the p-value according to the Wilcoxon test is $0.001 < P \leq 0.05$, and double plus or minus symbols denote high statistical significance with $P < 0.001$. Finally, when there is no statistically significant difference between the methods ($P > 0.05$), it is denoted by the symbol 0.

Results on BAliBASE 3.0

First, we evaluated the accuracy of PicXAA using the BAliBase 3.0 alignment benchmark. BAliBASE 3.0 is a widely used benchmark containing a total of 218 sequence alignments categorized into six reference sets. Two different criteria are used to score the alignment: the SP score, which is the percentage of the correctly aligned residue pairs in the alignment, and the *column score* (CS), which is the percentage of the correct columns in the alignment.

The SP and CS scores are given in Table 1. As shown in this table, PicXAA-PF outperforms other techniques both in terms of SP and CS scores on average. where this superiority is statistically significant in most cases. Moreover, we can see that, on average, all three versions of our algorithm (PicXAA-PF, PicXAA-PHMM and PicXAA-SPHMM) outperform their progressive counterparts (i.e., ProbAlign, ProbCons, and MUMMALS, respectively), which were used to estimate the alignment probabilities. This improvement is especially significant for the case of MUMMALS, where PicXAA-SPHMM improved the SP score by 1% and the CS score by 2.3%.

Further analysis of the alignment results indicates that the proposed alignment method yields the best (or close to the best) result for RV11, RV12, RV40 and RV50 reference sets, in terms of both SP and CS scores. Reference sets RV11 and RV12 contain equidistant families in which sequences that have large internal insertions (> 35 residues) are excluded. The sequence identity in RV11 is $<20\%$, while it is between 20% and 40% for RV12. As shown in the table, PicXAA has comparable performance to ProbAlign and ProbCons for these reference sets while it improves the performance of MUMMALS considerably. RV40 and RV50 contain sequences with large N/C-terminal extensions and large internal insertions, respectively. For these reference sets, alignment methods that can more effectively detect local similarities are expected to yield more accurate alignments. As we can see in Table 1, PicXAA yields the best performance on

Table 1. Performance evaluation on BAliBASE 3.0

Method	BALIBASE 3.0						
	RV11 SP/CS	RV12 SP/CS	RV20 SP/CS	RV30 SP/CS	RV40 SP/CS	RV50 SP/CS	Overall SP/CS
PicXAA-PF	68.92^{0}/**46.16***	94.61^{0}/86.20^{0}	92.49*/41.51*	86.11*/57.80*	**93.23***/**63.27***	89.23^{0}/53.00^{-}	**87.86***/**59.32***
PicXAA-PHMM	66.26^{-}/42.03^{-}	94.23^{-}/85.84^{0}	91.69^{-}/38.78^{0}	84.95^{0}/53.00^{-}	90.90^{--}/56.16^{--}	*90.16**/**60.19***	86.55^{--}/56.28^{--}
PicXAA-SPHMM	**69.46***/*44.74*^{0}	**94.79***/**86.36***	91.65^{0}/40.32^{0}	84.06^{0}/52.97^{0}	89.14^{-}/52.31^{-}	89.83^{0}/58.38^{0}	86.67^{0}/56.14^{0}
ProbAlign	*69.44*^{0}/44.53^{0}	*94.65*^{0}/*86.27*^{0}	*92.57*^{0}/43.93^{0}	85.29^{0}/56.57^{0}	92.22^{0}/60.35^{0}	88.96^{0}/54.94^{-}	*87.61*^{0}/58.82^{0}
ProbCons	66.87^{-}/41.61^{0}	94.10^{-}/85.48^{0}	91.68^{-}/40.63^{0}	84.55^{0}/54.37^{0}	90.59^{--}/54.63^{-}	88.96^{0}/55.88^{0}	86.42^{--}/56.01^{-}
MUMMALS	66.95^{-}/41.61^{0}	94.30^{0}/83.98^{0}	91.04^{0}/42.83^{0}	84.79^{0}/49.40^{-}	87.15^{-}/48.55^{-}	87.91^{0}/52.88^{0}	85.53^{-}/53.85^{-}
MAFFT-linsi	66.19^{-}/43.79^{0}	93.46^{--}/83.39^{--}	**92.70**^{0}/**45.12**^{0}	*86.80*$^{+}$/*59.33*^{0}	*92.61*$^{-}$/*61.51*$^{-}$	**90.25**^{0}/59.06^{0}	87.22^{-}/*59.28*^{0}
MAFFT-ginsi	65.69^{-}/42.66^{-}	93.16^{--}/83.41^{--}	92.48^{0}/41.85^{0}	85.92^{0}/55.23^{0}	91.15^{-}/56.24^{-}	89.95^{0}/59.25^{0}	86.56^{--}/56.73^{--}
MAFFT-einsi	66.01^{-}/43.74^{0}	93.45^{--}/83.39^{--}	92.51^{0}/*44.32*^{0}	**86.81**$^{+}$/**59.43**^{0}	92.26^{--}/60.53^{-}	89.87^{0}*59.63*^{0}	87.05^{--}/58.95^{--}
MAFFT-fftnsi	61.45^{--}/39.45^{-}	90.82^{--}/79.57^{--}	90.89^{--}/38.24^{0}	83.22^{-}/49.00^{-}	89.88^{--}/54.67^{--}	86.41^{--}/53.38^{--}	84.13^{--}/53.08^{--}
T-Coffee	65.98^{--}/41.37^{--}	94.07^{--}/85.25^{--}	91.49^{-}/38.71^{0}	83.71^{-}/49.47^{-}	89.69^{--}/55.12^{--}	89.41^{0}/58.06^{0}	85.94^{--}/55.16^{--}
DIALIGN-TX	51.52^{--}/26.53^{--}	89.18^{--}/75.23^{--}	87.87^{--}/30.49^{--}	76.18^{--}/38.53^{--}	83.64^{--}/44.82^{--}	82.28^{--}/46.56^{--}	78.83^{--}/44.33^{--}
MUSCLE	57.16^{--}/31.79^{--}	91.53^{--}/80.39^{--}	88.90^{--}/35.00^{-}	81.44^{--}/40.87^{--}	86.48^{--}/45.02^{--}	83.53^{--}/45.94^{--}	81.94^{--}/47.46^{--}
CLUSTALW	49.43^{--}/23.95^{--}	87.14^{--}/71.86^{--}	86.16^{--}/23.46^{--}	71.97^{--}/26.90^{--}	78.62^{--}/40.00^{--}	73.36^{--}/30.38^{-}	75.37^{--}/38.01^{--}

these reference sets, which shows that constructing the alignment from confidently alignable regions with high local similarities leads to more accurate results in such cases.

In RV20, each alignment consists of a family with $> 40\%$ similarity and an orphan sequence that shares $< 20\%$ identity to other sequences. RV30 contains subfamilies with $> 40\%$ identity within each subfamily while having $< 20\%$ similarity between the sequences of different subfamilies. Since the alignments in RV20 and RV30 consist of sequences that belong to distantly related subfamilies, we would expect the progressive approach to perform better, since it first aligns the homologous sequences in each subfamily and then align the sequence profiles of the respective families. In fact, MAFFT-(le)insi shows the best performance on these reference sets, and the high CS scores of various progressive algorithms demonstrate their advantage on such sets. However, Wilcoxon test shows that the superiority of the progressive techniques over PicXAA on these reference sets is mostly not statistically significant.

Results on IRMBASE 2.0

To assess the effectiveness of PicXAA in capturing local similarities among sequences, we carried out another experiment based on the IRMBASE 2.0 alignment benchmark. IRMBASE 2.0 has been constructed by inserting highly conserved motifs generated by ROSE (39) in long random sequences. IRMBASE 2.0 consists of four reference sets with a total of 192 alignments. Similar to BAliBASE 3.0, we use the SP and CS scores to evaluate the alignment quality.

The average SP and CS scores of the tested methods are given in the first column of Table 2. The SP and CS scores for different reference sets of IRMBASE 2.0 can be found in Supplementary Table S1. As we can see in these tables, DIALIGN-TX shows the best overall performance on IRMBASE 2.0. However, it must be noted that IRMBASE 2.0 was originally developed to evaluate the performance of DIALIGN-TX to align sequences with local similarities, hence DIALIGN-TX may have been especially fine-tuned to perform well on this data set. In general, DIALIGN-TX does not perform well on other alignment benchmarks (Tables 1 and 2). Besides, we can see that even on IRMBASE 2.0, the superiority of DIALIGN-TX over PicXAA is not statistically significant for the SP score. Except for DIALIGN-TX, we can see that PicXAA yields the best overall performance among all other alignment methods with statistically significant superiority in most cases.

Comparing the three implementations of PicXAA with their progressive counterparts (ProbAlign, ProbCons and MUMMALS) shows that the proposed alignment method leads to significant improvement (with $P \leq 10^{-14}$) in the overall accuracy for all reference sets. On average, the SP score is improved by 5–8% and the CS score is improved by 8–13%. These results clearly demonstrate the strength of PicXAA in picking up local similarities among the aligned sequences that are often missed by progressive methods. Table 2 also shows that MUMMALS, T-Coffee, MUSCLE, and CLUSTALW do not perform well on IRMBASE 2.0. In terms of the SP score, MAFFT-einsi performs well on IRMBASE 2.0 (with no statistically significant superiority over PicXAA-PHMM), where PicXAA-PHMM yields comparable results. However, PicXAA-PHMM results in significantly higher CS score in all the reference sets (with $P \leq 10^{-5}$), which is on average 6.2% higher than that of the runner-up (i.e. MAFFT-einsi).

Results on SABmark 1.65

We also evaluated the performance of PicXAA on SABmark 1.65. This dataset includes two sets of reference alignments derived from the SCOP classification (40): 'Twilight Zone' and 'Superfamilies'. For each alignment, the collection of pairwise reference structural alignments is provided. The accuracy of the alignment is measured using the f_D score, an equivalent of the SP score.

Table 2. Performance evaluation on IRMBASE 2.0, SABmark 1.65, PREFAB 4.0, HOMSTRAD, and OXBench 1.3

Method	IRMBASE 2.0	SABmark 1.65		PREFAB 4.0	HOMSTRAD	OXBench 1.3		
	Overall SP/CS	Twilight f_D/f_M	Superfamily f_D/f_M	Q	SP/CS	master SP/CS	full SP/CS	extended SP/CS
PicXAA-PF	89.00^{--}/50.08^{--}	16.75^{--}/15.37^{--}	49.66^{--}/41.41^{--}	71.34^{--}	82.36^{--}/7813^{--}	89.72^{--}/84.72^{--}	**84.15***/**76.46***	92.41^{--}/88.08^{--}
PicXAA-PHMM	90.76*/*54.48**	17.12^{--}/14.65^{--}	50.37^{--}/41.13^{--}	71.16^{--}	82.08^{--}/77.74^{--}	89.28^{--}/84.05^{--}	83.23^{--}/75.16^{--}	92.10^{--}/87.65^{--}
PicXAA-SPHMM	72.75^{--}/33.02^{--}	**20.99***/*17.12**	**53.53***/**42.77***	*72.38**	**84.04***/**80.04***	**90.55***/**85.68***	83.78^{--}/75.96^{--}	*92.99**/88.86*
ProbAlign	81.68^{--}/36.69^{--}	15.86^{--}/13.05^{--}	48.66^{--}/39.82^{--}	71.90^{--}	82.39^{--}/78.15^{--}	89.79^{--}/84.93^{--}	*84.07*^{0}/*76.42*^{0}	92.55^{0}/88.41^{0}
ProbCons	85.30^{--}/42.51^{--}	16.64^{--}/13.55^{--}	48.56^{--}/39.51^{--}	71.64^{--}	82.04^{--}/77.74^{--}	89.29^{--}/84.10^{--}	83.25^{--}/75.18^{--}	92.43^{--}/88.15^{--}
MUMMALS	68.44^{--}/24.62^{--}	*19.99*$^{--}$/**18.23**^{0}	*52.08*$^{--}$/*42.74*$^{--}$	**72.73**^{0}	*83.79*$^{--}$/*79.66*$^{--}$	*90.20*$^{--}$/*85.21*$^{--}$	82.81^{--}/75.05^{--}	92.52^{--}/87.79^{--}
MAFFT-linsi	89.44^{-}/46.02^{--}	17.42^{--}/13.16^{--}	50.47^{--}/40.01^{--}	72.16^{--}	80.31^{--}/75.78^{--}	88.30^{--}/82.79^{--}	82.83^{--}/74.74^{--}	92.95^{0}/88.92^{0}
MAFFT-ginsi	84.53^{--}/40.24^{--}	17.40^{--}/13.13^{--}	50.23^{--}/39.76^{--}	71.65^{--}	80.20^{--}/75.60^{--}	88.71^{--}/83.27^{--}	81.91^{--}/73.50^{--}	**93.19**^{0}/**89.14**^{0}
MAFFT-einsi	*91.77*^{0}/48.21^{--}	17.77^{--}/13.07^{--}	49.94^{--}/39.23^{--}	72.04^{--}	80.32^{--}/75.80^{--}	88.32^{--}/82.83^{--}	82.98^{--}/74.87^{--}	*92.99*^{0}/*88.97*^{0}
MAFFT-fftnsi	82.76^{--}/38.51^{--}	11.54^{--}/9.04^{--}	42.43^{--}/34.89^{--}	68.78^{--}	78.31^{--}/73.51^{--}	87.69^{--}/82.02^{--}	81.50^{--}/73.07^{--}	91.35^{--}/86.83^{--}
T-Coffee	87.75^{--}/46.33^{--}	15.18^{--}/11.88^{--}	42.47^{--}/34.20^{--}	70.75^{--}	76.79^{--}/71.97^{--}	81.78^{--}/75.04^{--}	73.38^{--}/63.75^{--}	91.28^{--}/86.64^{--}
DIALIGN-TX	**92.92**^{0}/**71.02**$^{++}$	11.01^{--}/11.59^{--}	39.30^{--}/35.28^{--}	62.47^{--}	76.85^{--}/71.53^{--}	85.96^{--}/79.65^{--}	80.76^{--}/72.26^{--}	88.79^{--}/82.89^{--}
MUSCLE	43.73^{--}/11.12^{--}	12.91^{--}/8.94^{--}	43.07^{--}/33.56^{--}	68.26^{--}	80.91^{--}/76.36^{--}	89.25^{--}/84.07^{--}	82.46^{--}/74.20^{--}	91.61^{--}/87.01^{--}
CLUSTALW	26.34^{--}/2.44^{--}	12.87^{--}/8.72^{--}	38.62^{--}/30.27^{--}	61.75^{--}	80.14^{--}/75.42^{--}	89.29^{--}/83.79^{--}	81.56^{--}/72.66^{--}	89.40^{--}/83.86^{--}

An additional f_M score is also computed in SABmark 1.65 to measure the specificity of the alignment.

As we can see in Table 2, PicXAA-SPHMM yields the highest f_D score with statistically significant superiority on both the Twilight and Superfamily sets. In terms of specificity (f_M score), PicXAA-SPHMM performs best on the Superfamily set, while MUMMALS outperforms other methods on the Twilight reference set. However, the superiority of MUMMALS over PicXAA-SPHMM on the Twilight set (in terms of f_M score) is not statistically significant, while the superiority of PicXAA-SPHMM (in terms of f_D score) is statistically significant ($P \leq 0.001$). We can observe that the use of structural pair-HMM leads to more accurate alignments results for PicXAA-SPHMM and MUMMALS on this benchmark, owing to more accurate estimation of the pairwise residue alignment probabilities. PicXAA-PHMM and PicXAA-PF, which do not incorporate structural information in estimating the probabilities, yield lower f_D and f_M scores compared PicXAA-SPHMM. However, both PicXAA-PHMM and PicXAA-PF significantly outperform their progressive counterparts (i.e. ProbCons and ProbAlign), and we obtain 1–2% improvement in f_D and f_M scores.

Results on PREFAB 4.0

Here, we evaluated PicXAA on the PREFAB 4.0 alignment benchmark. PREFAB 4.0 (21) consists of 1682 alignments. Each alignment contains one pair of sequences with known 3D structure and additional sequences that are obtained from high scoring hits of these two sequences in a database search.

Table 2 shows the average Q-score over the 1682 alignments in PREFAB 4.0 for every alignment method. For each alignment, the Q-score is computed according to the 3D structural alignment of the core sequence pair contained in the alignment. Therefore, it would be natural to expect that MUMMALS, which employs structural information for computing the alignment probabilities, will perform well on PREFAB 4.0 as on SABmark 1.65. This can be observed in Table 2, where MUMMALS shows best performance among all the compared methods. However, the performance difference between MUMMALS and PicXAA-SPHMM is not statistically significant. For all other methods, PicXAA-SPHMM shows significantly better performance (with $P \leq 0.001$) in terms of the Q-score.

Table 2 also shows that PicXAA-PF, PicXAA-PHMM and PicXAA-SPHMM yield slightly lower average Q-scores compared with the corresponding progressive methods, ProbAlign, ProbCons and MUMMALS. One possible explanation for this observation can be found from the benchmark generation process. As mentioned earlier, each alignment in PREFAB 4.0 consists of a pair of protein sequences with known 3D structure and other homologous sequences obtained from database search. For this reason, every alignment in the PREFAB 4.0 basically consists of two groups of similar sequences, or two subfamilies. When using the progressive alignment technique, the guide tree will separate the sequences in the two subfamilies, hence the algorithm will first align the sequences in respective families and align the two subfamilies at the end. As noted in (41), recruiting homologous sequences via database search and performing a profile–profile alignment is known to improve the alignment accuracy compared with pairwise alignment of individual sequences, which explains why the progressive techniques slightly outperform PicXAA on PREFAB 4.0.

Results on HOMSTRAD

We performed a comparison based on HOMSTRAD (37) alignment benchmark (dated 1 January 2010), containing 1031 homologous protein families aligned based on their 3D structures. Table 2 shows the average SP and CS scores. We can see that PicXAA-SPHMM performed best among all the compared methods, with high statistical significance. In fact, PicXAA-SPHMM outperformed the runner-up (i.e. MUMMALS) with a $P \leq 10^{-5}$.

As shown in Supplementary Table S2, the performance of PicXAA on HOMSTARD stands out even more when we consider only the alignments with >3 sequences (totally 233 alignments). In this case, PicXAA-SPHMM outperforms MUMMALS by 0.65% in the SP and 1.12% in the CS score.

Results on OXBench 1.3

Finally, we used OXBench 1.3 (38) benchmark, a set of structure-based alignments of protein domains. OXBench consists of three reference sets: 'master', a set of 672 reference alignments of protein structural domains; 'full', a set of 605 full-length sequences of the domains in the master dataset; and 'extended', the master set of domains augmented by sequences of unknown structure. We excluded seven large alignments in the extended reference set due to running time consideration. Last three columns of Table 2 report the SP and CS scores.

We can see that in both scores, PicXAA-SPHMM and PicXAA-PF outperform all other techniques for the master and full reference sets, respectively. We can also see that, in most cases, this superiority is statistically significant with $P \leq 10^{-4}$. For the extended set, MAFFT-[lge]insi shows the best performance, while PicXAA-SPHMM yields the next best result. However, this inferiority is not statistically significant, and PicXAA-SPHMM significantly outperforms most of the other methods.

Expected number of correctly aligned residues

The primary goal of PicXAA is to find the MSA that maximizes the expected accuracy, or the expected number of correctly aligned residues. To assess the effectiveness of PicXAA in achieving this goal, we introduce the 'sum-of-pairwise probabilities' (SPP) score. The SPP score computes the total expected number of correct pairwise alignments, and it can serve as a good indicator of the expected accuracy of a predicted alignment. The SPP score is computed as

$$\mathrm{SPP} = \sum_{\mathrm{x,y} \in \mathrm{S}} \sum_{x_i \sim y_j \in a} P(x_i \sim y_j | \mathrm{x,y}),$$

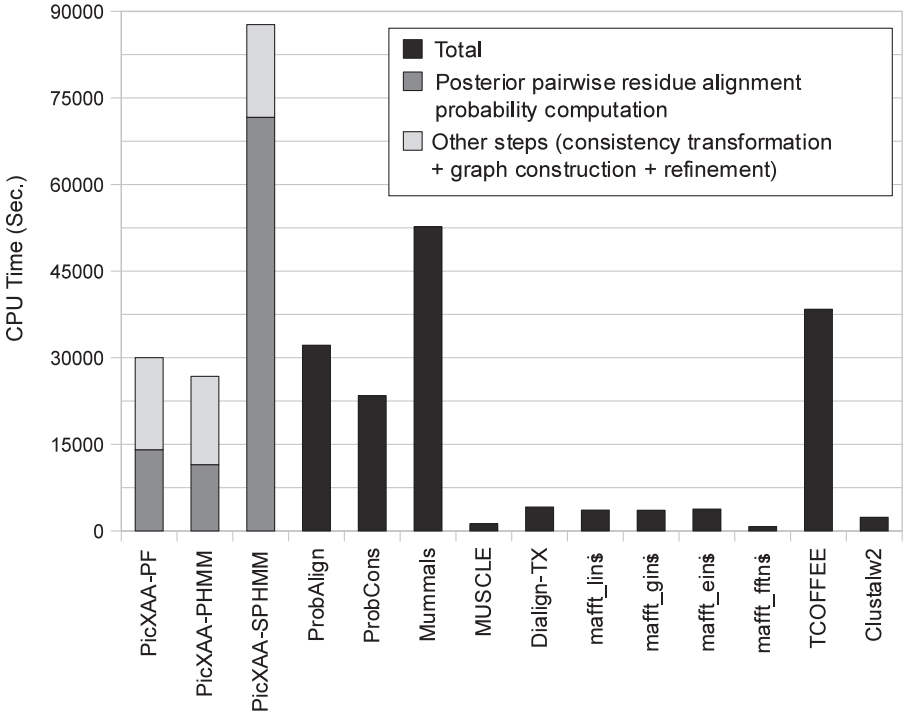

Figure 2. Total CPU time for aligning the sequences in BAliBASE 3.0 (shown in seconds). For different implementations of PicXAA, the total CPU time is divided into two parts: the time that takes for probability estimation and the time needed in the remaining steps for constructing the alignment.

by summing up the posterior pairwise alignment probabilities of all aligned residue pairs in the final alignment *a*. It is important to remember that the original (non-transformed) probabilities should be used to compute the SPP score, since the transformed probabilities do not directly reflect the pairwise residue alignment probabilities. In fact, the consistency transformation mainly aims to adjust the original pairwise alignment probabilities to obtain a consistent, hence more accurate, MSA.

In this experiment, we computed the sum of SPP scores for all alignments in BAliBASE 3.0 and IRMBASE 2.0 for two versions of the proposed alignment method, PicXAA-PF and PicXAA-PHMM, and compared the results with their progressive counterparts, i.e. ProbAlign and ProbCons, respectively. We did not compare PicXAA-SPHMM and MUMMALS, since MUMMALS does not compute all pairwise residue alignment probabilities, in general. As shown in Supplementary Table S3, the PicXAA slightly improves the SPP score compared with other progressive methods, when evaluated on BAliBASE 3.0. This improvement is especially significant for the RV30 reference set, which consists of distantly related subfamilies. Evaluation on IRMBASE 2.0 shows that PicXAA remarkably improves the SPP score over the progressive methods, where we improve the score by 9.5% compared with ProbAlign and 3.2% compared with ProbCons. These results demonstrate the effectiveness of the probabilistic greedy construction scheme, adopted by PicXAA, in enhancing the expected accuracy of the MSA, especially for sequences with only local similarities.

Computational complexity

As shown in Figure 2, PicXAA-PF and PicXAA-PHMM has comparable speed with their progressive counterparts, ProbAlign and ProbCons. This implies that we can obtain more accurate MSAs, which effectively capture the local similarities among sequences, by using the proposed probabilistic greedy alignment approach without any substantial increase in the overall computational complexity compared with the conventional progressive approach. Further discussion on the computational complexity of PicXAA can be found in Supplementary Data.

DISCUSSION

Overall performance of PicXAA

To find the MEA alignment for multiple sequences, we developed PicXAA, a probabilistic non-progressive algorithm that greedily builds up the alignment by successively adding the most probable residue pair to the MSA. For fast construction of the multiple sequence alignment, we took a graph-based approach as in (32,33), in which the overall compatibility of the alignment is verified and maintained using efficient graph-based techniques. We also introduced a novel consistency transformation and a discriminative refinement technique, which can altogether improve the accuracy of the final MSA when combined with the probabilistic greedy alignment technique.

As demonstrated in our results, PicXAA consistently shows excellent performance on various reference sets with different characteristics, from reference sets containing sequences that belong to closely related families,

to reference sets containing sequences that belong to distant families, hence share only local similarities. In other words, PicXAA is not fine-tuned nor overtrained to perform well on specific types of sequence sets, and it can yield accurate alignment results under various circumstances. However, its advantage may be most clearly seen on datasets with only local similarities, where progressive techniques fail to capture those similarities, while PicXAA effectively grasps such similarities through the proposed probabilistic greedy alignment approach.

Experimental results on IRMBASE 2.0 clearly show PicXAA's strength in grasping local similarities among distantly related sequences. In fact, the proposed method exhibits a significant improvement over various progressive alignment techniques, and it is outperformed only by DIALIGN-TX (not statistically significant in terms of the SP score) on this dataset, which does not generally perform well on reference sets that contain sequences with global similarities. As shown in Tables 1 and 2, PicXAA also yields accurate results on other benchmarks such as BAliBASE 3.0, SABmark 1.65, PREFAB 4.0, HOMSTRAD and OXBench 1.3. PicXAA has both the highest average SP and CS scores on BAliBASE 3.0, HOMSTRAD and OXBENCH 1.3, and it results in the best overall performance, in terms of average f_d (sensitivity) and f_M (specificity) scores, on SABmark 1.65.

The statistical significance test also demonstrates that PicXAA consistently outperforms other alignment techniques on various benchmark datasets. When PicXAA yields a higher average score compared with another algorithm, this superiority is statistically significant in most cases (usually with very low *P*-values). However, when another technique outperforms PicXAA, this performance difference is often not statistically significant (except for the CS score of DIALIGN-TX on IRMBASE 2.0).

Furthermore, Supplementary Table S3 shows that PicXAA can effectively increase the number of correctly aligned residues, hence capable of finding more accurate MSAs according to the MEA criterion, especially for sequence sets with local similarities.

Proposed consistency transformation improves the alignment accuracy

The new probabilistic consistency transformation updates the residue alignment probabilities in a given sequence pair, by using the information from other sequences in the alignment according to their relation to the given pair. When we have divergent sequences with only local similarities or a set of sequences that belong to distantly related subfamilies, this technique can improve the overall quality of the final MSA, by incorporating only the information from related sequences in the transformation. In fact, the conventional consistency transformation tends to water down local similarities that are shared only by a subset of sequences, thereby breaking up related residues into different columns.

To assess the effectiveness of the proposed transformation, we conducted the following experiment. In this test, we examined the performance improvement that can be achieved by incorporating the new consistency transformation in two popular probabilistic consistency-based alignment algorithms, ProbAlign (27) and ProbCons (20). Figure 3A and B summarize the average SP and CS scores on IRMBASE 2.0 and BAliBASE 3.0 benchmarks. As we can observe in these figures, the new consistency transformation significantly improves the SP and CS scores of both algorithms on the IRMBASE 2.0 dataset. The improvement is near 1.5% in the SP score and 3.3% in the CS score for ProbAlign, and 1.6% in the

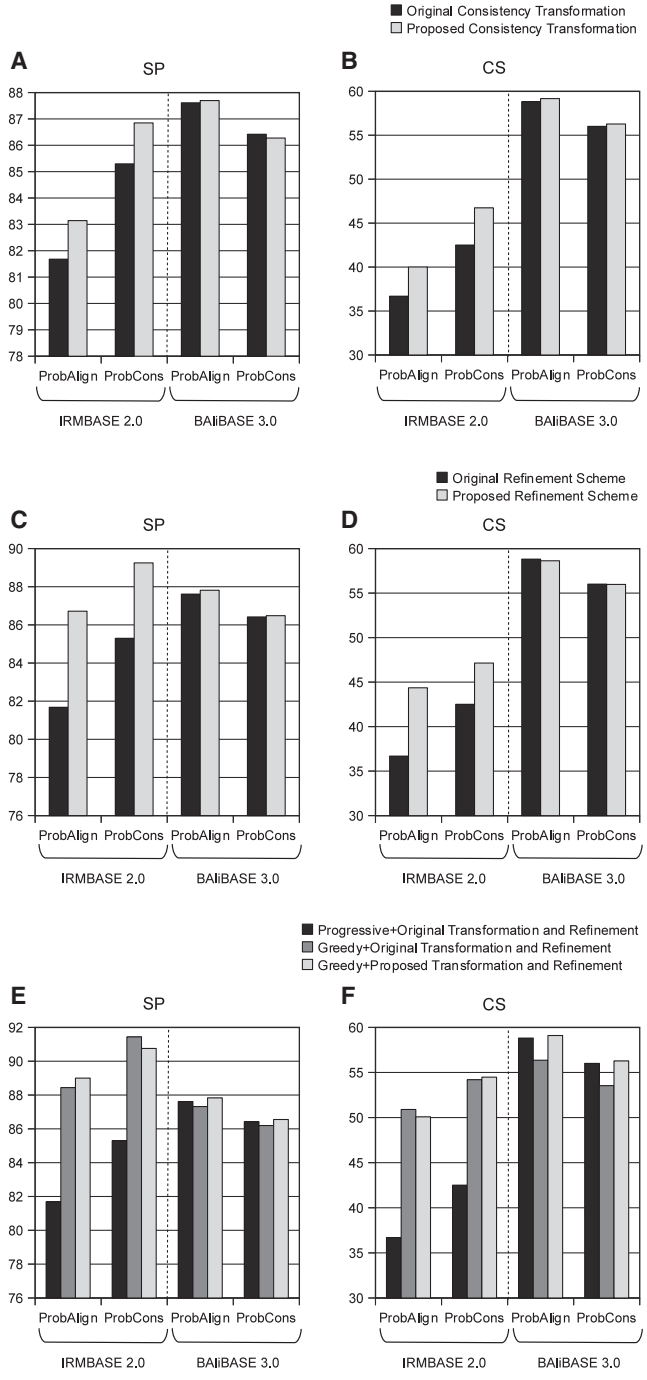

Figure 3. Effectiveness of the proposed techniques. (**A** and **B**) Novel consistency transformation; (**C** and **D**) Discriminative refinement strategy; (**E** and **F**) Greedy graph-based alignment.

SP score and 4.2% in the CS score for ProbCons. This clearly shows the advantage of the proposed probabilistic consistency transformation over the conventional transformation on reference sets with local similarities. For BAliBASE 3.0, the proposed transformation does not result in a significant change in average the SP score, but it leads to about 0.3% improvement in the average CS score. The improvement is especially significant for the reference set RV30, which consists of sequences that belong to diverse subfamilies, where we have 1.6% (0.1%) improvement in the SP score and 3.6% (1.7%) improvement in the CS score for ProbAlign (ProbCons).

Discriminative refinement improves the alignment quality in low confidence regions

Residues with low alignment probability are difficult to correctly align. In low similarity regions, the profile–profile alignment approach can help improve the overall alignment accuracy. In this work, we proposed a discriminative refinement technique, which divides the alignment into two sequence groups based on sequence similarity and realigns them in an iterative manner (see 'Methods' section). For a given MSA, this technique can improve the alignment quality in low confidence regions, while preserving residues pairs that have been confidently aligned.

To investigate the effectiveness of the proposed refinement strategy, we replaced the conventional refinement scheme used in ProbAlign and ProbCons by the proposed scheme, and analyzed the performance changes. The average SP and CS scores are shown in Figure 3C and D. Experimental results on IRMBASE 2.0 demonstrate that the new refinement strategy leads to considerable improvement over the conventional refinement strategy. This improvement is about 5% in the SP score and 7.7% in the CS score for ProbAlign, and 4% in the SP score and 4.6% in the CS score for ProbCons. Similar experiments on BAliBASE 3.0 did not result in significant changes, as we can see in Figure 3C and D. These results show that the 'discriminative'-split-and-realignment technique proposed in this article is especially effective for improving the alignment accuracy in sequence sets with local similarities. In fact, we could observe that the conventional refinement strategy tends to break many confidently aligned residue pairs in such datasets, due to the 'random'-split-and-realignment process.

Greedy alignment construction leads to more accurate alignments

The proposed probabilistic greedy alignment technique, which adds a single residue pair with the highest alignment probability at each step, has a crucial positive impact on the overall alignment accuracy. In fact, remarkable performance gain can be obtained by considering all possible residue pairs between all sequence pairs simultaneously, and constructing the alignment by adding the most probable residue pairs one after another. By building up the MSA from the confidently alignable regions, PicXAA significantly reduces the risk of propagating the alignment errors made at the early stage to the final alignment, which is a commonly observed problem in many progressive algorithms. Furthermore, the proposed alignment scheme is also effective in maximizing the number of correctly aligned residues, as can be observed in Supplementary Table S3.

To analyze the effect of the proposed probabilistic greedy alignment approach on the alignment quality, we examined the performance change of ProbAlign and ProbCons after replacing the progressive alignment scheme with the proposed greedy alignment scheme. Figure 3E and F show the average SP and CS scores of the different alignment methods. For IRMBASE 2.0, we can see a remarkable enhancement in both the SP and CS scores that results from replacing the progressive scheme with the proposed greedy scheme, while retaining the conventional consistency transformation and refinement technique. The resulting enhancement is about 6.8% in the SP score and 14.2% in the CS score for ProbAlign, and 6.1% in the SP score and 11.7% in the CS score for ProbCons. Experiments based on BAliBASE 3.0 show that the greedy alignment scheme alone, without employing the proposed consistency transformation and the discriminative refinement technique, does not necessarily improve the overall accuracy of the alignment. This is particularly evident when we look at the CS score, where we see 2–3% reduction. This degradation is mainly due to the lower performance on the reference sets RV20 and RV30, where each alignment consists of divergent subfamilies. (In the case of RV20, each alignment contains one orphan sequence that bears low similarity with the rest of the sequences.) As discussed earlier, the conventional progressive alignment technique is expected to work well on such reference sets. However, combination of the greedy alignment scheme with the new consistency transformation and refinement strategy effectively overcomes these problems by considering both the intra- and inter-family similarities. We can see in Figure 3E and F that the greedy approach integrated with the proposed consistency transformation and the new refinement technique improves the alignment accuracy. Such a phenomenon is not observed in experiments based on IRMBASE 2.0, since IRMBASE 2.0 does not contain reference sets with different subfamilies.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their helpful remarks.

FUNDING

Funding for open access charge: Texas A&M faculty start up fund.

Conflict of interest statement. None declared.

REFERENCES

1. Phillips,A., Janies,D. and Wheeler,W. (2000) Multiple sequence alignment in phylogenetic analysis. *Mol. Phylogenet. Evol.*, **16**, 317–330.
2. Wong,K.M., Suchard,M.A. and Huelsenbeck,J.P. (2008) Alignment uncertainty and genomic analysis. *Science*, **319**, 473–476.
3. Cuff,J.A. and Barton,G.J. (2000) Application of multiple sequence alignment profiles to improve protein secondary structure prediction. *Proteins*, **40**, 502–511.
4. Kemena,C. and Notredame,C. (2009) Upcoming challenges for multiple sequence alignment methods in the high-throughput era. *Bioinformatics*, **25**, 2455–2465.
5. Edgar,R.C. and Batzoglou,S. (2006) Multiple sequence alignment. *Curr. Opin. Struct. Biol.*, **16**, 368–373.
6. Pei,J. (2008) Multiple protein sequence alignment. *Curr. Opin. Struct. Biol.*, **18**, 382–386.
7. Kumar,S. and Filipski,A. (2007) Multiple sequence alignment: in pursuit of homologous DNA positions. *Genome Res.*, **17**, 127–135.
8. Notredame,C. (2002) Recent progress in multiple sequence alignment: a survey. *Pharmacogenomics*, **3**, 131–144.
9. Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, **48**, 443–453.
10. Smith,T.F. and Waterman,M.S. (1981) Identification of common molecular subsequences. *J. Mol. Biol.*, **147**, 195–197.
11. Wang,L. and Jiang,T. (1994) On the complexity of multiple sequence alignment. *J. Comput. Biol.*, **1**, 337–348.
12. Gondro,C. and Kinghorn,B.P. (2007) A simple genetic algorithm for multiple sequence alignment. *Genet. Mol. Res.*, **6**, 964–982.
13. Riaz,T., Yi,W. and Li,K.B. (2005) A tabu search algorithm for post-processing multiple sequence alignment. *J. Bioinform. Comput. Biol.*, **3**, 145–156.
14. Lenhof,H.-P., Reinert,K. and Vingron,M. (1998) A polyhedral approach to RNA sequence alignment. *Proceedings of the Second Annual International Conference on Computational Molecular Biology (RECOMB-98)*. ACM Press, New York, NY, USA, pp. 153–162.
15. Eddy,S.R. (1995) Multiple alignment using hidden Markov models. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **3**, 114–120.
16. Hogeweg,P. and Hesper,B. (1984) The alignment of sets of sequences and the construction of phyletic trees: an integrated method. *J. Mol. Evol.*, **20**, 175–186.
17. Feng,D.F. and Doolittle,R.F. (1987) Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J. Mol. Evol.*, **25**, 351–360.
18. Thompson,J.D., Higgins,D.G. and Gibson,T.J. (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.
19. Notredame,C., Higgins,D.G. and Heringa,J. (2000) T-Coffee: a novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.*, **302**, 205–217.
20. Do,C.B., Mahabhashyam,M.S., Brudno,M. and Batzoglou,S. (2005) ProbCons: probabilistic consistency-based multiple sequence alignment. *Genome Res.*, **15**, 330–340.
21. Edgar,R.C. (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.*, **32**, 1792–1797.
22. 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.
23. Katoh,K., Kuma,K., Toh,H. and Miyata,T. (2005) MAFFT version 5: improvement in accuracy of multiple sequence alignment. *Nucleic Acids Res.*, **33**, 511–518.
24. Katoh,K. and Toh,H. (2008) Recent developments in the MAFFT multiple sequence alignment program. *Brief. Bioinform.*, **9**, 286–298.
25. Pei,J. and Grishin,N.V. (2006) MUMMALS: multiple sequence alignment improved by using hidden Markov models with local structural information. *Nucleic Acids Res.*, **34**, 4364–4374.
26. Paten,B., Herrero,J., Beal,K. and Birney,E. (2009) Sequence progressive alignment, a framework for practical large-scale probabilistic consistency alignment. *Bioinformatics*, **25**, 295–301.
27. Roshan,U. and Livesay,D.R. (2006) Probalign: multiple sequence alignment using partition function posterior probabilities. *Bioinformatics*, **22**, 2715–2721.
28. Rausch,T., Emde,A.K., Weese,D., Dring,A., Notredame,C. and Reinert,K. (2008) Segment-based multiple sequence alignment. *Bioinformatics*, **24**, i187–i192.
29. Morgenstern,B., Frech,K., Dress,A. and Werner,T. (1998) DIALIGN: finding local similarities by multiple sequence alignment. *Bioinformatics*, **14**, 290–294.
30. Subramanian,A.R., Weyer-Menkhoff,J., Kaufmann,M. and Morgenstern,B. (2005) DIALIGN-T: an improved algorithm for segment-based multiple sequence alignment. *BMC Bioinformatics*, **6**, 66.
31. Subramanian,A.R., Kaufmann,M. and Morgenstern,B. (2008) DIALIGN-TX: greedy and progressive approaches for segment-based multiple sequence alignment. *Algorithms Mol. Biol.*, **3**, 6.
32. Schwartz,A.S. and Pachter,L. (2007) Multiple alignment by sequence annealing. *Bioinformatics*, **23**, e24–e29.
33. Bradley,R.K., Roberts,A., Smoot,M., Juvekar,S., Do,J., Dewey,C., Holmes,I. and Pachter,L. (2009) Fast statistical alignment. *PLoS Comput. Biol.*, **5**, e1000392.
34. Lu,Y. and Sze,S.H. (2009) Improving accuracy of multiple sequence alignment algorithms based on alignment of neighboring residues. *Nucleic Acids Res.*, **37**, 463–472.
35. Thompson,J.D., Koehl,P., Ripp,R. and Poch,O. (2005) BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, **61**, 127–136.
36. Van Walle,I., Lasters,I. and Wyns,L. (2005) SABmark—a benchmark for sequence alignment that covers the entire known fold space. *Bioinformatics*, **21**, 1267–1268.
37. Mizuguchi,K., Deane,C.M., Blundell,T.L. and Overington,J.P. (1998) HOMSTRAD: a database of protein structure alignments for homologous families. *Protein Sci.*, **7**, 2469–2471.
38. Raghava,G.P., Searle,S.M., Audley,P.C., Barber,J.D. and Barton,G.J. (2003) OXBench: a benchmark for evaluation of protein multiple sequence alignment accuracy. *BMC Bioinformatics*, **4**, 47.
39. Stoye,J., Evers,D. and Meyer,F. (1998) Rose: generating sequence families. *Bioinformatics*, **14**, 157–163.
40. Murzin,A.G., Brenner,S.E., Hubbard,T. and Chothia,C. (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.
41. Altschul,S.F., Madden,T.L., Schffer,A.A., Zhang,J., Zhang,Z., Miller,W. and Lipman,D.J. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.