BIOINFORMATICS ORIGINAL PAPER Vol. 27 no. 1 2011, pages 31–37
doi:10.1093/bioinformatics/btq621

Sequence analysis Advance Access publication November 24, 2010

A performance enhanced PSI-BLAST based on hybrid alignment

Yuheng Li[1], Nicholas Chia[2,3], Mario Lauria[4] and Ralf Bundschuh[5,6,7,*]

[1]Covidien, 60 Middletown Avenue, North Haven, CT, 06473, [2]Institute for Genomic Biology, 1206 West Gregory Drive, [3]Department of Physics, University of Illinois at Urbana-Champaign, 1110 West Green Street, Urbana, IL 61801, USA, [4]Systems Biology Group, Telethon Institute of Genetics and Medicine (TIGEM), via P. Castellino 111, 80131 Naples, Italy, [5]Department of Physics, Biophysics Graduate Program, The Ohio State University, 191 West Woodruff Avenue, [6]Department of Biochemistry, 484 West 12th Avenue and [7]Center for RNA Biology, 318 West 12th Avenue, Columbus, OH 43210, USA

Associate Editor: John Quackenbush

ABSTRACT

Motivation: Sequence alignment is one of the most popular tools of modern biology. NCBI's PSI-BLAST utilizes iterative model building in order to better detect distant homologs with greater sensitivity than non-iterative BLAST. However, PSI-BLAST's performance is limited by the fact that it relies on deterministic alignments. Using a semi-probabilistic alignment scheme such as Hybrid alignment should allow for better informed model building and improved identification of homologous sequences, particularly remote homologs.

Results: We have built a new version of the tool in which the Smith-Waterman alignment algorithm core is replaced by the hybrid alignment algorithm. The favorable statistical properties of the hybrid algorithm allow the introduction of position-specific gap penalties in Hybrid PSI-BLAST. This improves the position-specific modeling of protein families and results in an overall improvement of performance.

Availability: Source code is freely available for download at http://bioserv.mps.ohio-state.edu/HybridPSI, implemented in C and supported on linux.

Contact: bundschuh@mps.ohio-state.edu

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

Received on August 23, 2010; revised on October 20, 2010; accepted on October 29, 2010

1 INTRODUCTION

Sequence alignment is one of the most commonly used tools in modern biology. The BLAST (Altschul *et al.*, 1990) and PSI-BLAST (Altschul *et al.*, 1997) algorithms have become standard tools for homology detection. In particular, PSI-BLAST (Altschul *et al.*, 1997) iteratively refines models for a protein family of interest with the goal of working toward the detection of weaker or more distant homologs. The detection of deeper evolutionary relationships can impact many aspects of biology such as the broadness of evolutionary studies (Dokholyan and Shakhnovich, 2001), the quality of gene annotations (Bateman *et al.*, 2002; Schwede *et al.*, 2003), protein structure/function predictions (Jones, 1999; Zhou and Zhou, 2005) and our understanding of genes found in metagenomic surveys (Rondon *et al.*, 2000). Improving PSI-BLAST has the potential to affect a large number of users and scientific applications.

The idea behind PSI-BLAST, or Position Specific Iterated Basic Local Alignment Search Tool, is to leverage information from identified homologs to create a position-specific substitution matrix that can then be used to detect even more distant homologs (Altschul *et al.*, 1997). These searches take place in an iterative fashion by using homologs identified in the previous round for the model building phase where the position-specific substitution matrix is refined for the next round. While this method increases the sensitivity of the subsequent searches, the BLAST core imposes a severe limitation—namely, an inability to incorporate position-specific gap costs. This limitation stems from the inability to statistically assess the significance of alignment scores with position-specific gap costs on practical timescales.

Here, we overcome this limitation by replacing the alignment core of PSI-BLAST by the semi-probabilistic hybrid alignment algorithm (Yu and Hwa, 2001). This algorithm's main advantage over the alignment core of NCBI PSI-BLAST is its statistical tractability, particularly with regards to position-specific gap costs (Yu *et al.*, 2002). In effect, Hybrid alignment introduces to PSI-BLAST one of the advantages usually reserved for profile Hidden Markov Models (Eddy, 1998, 2008; Krogh *et al.*, 1994). The Hybrid PSI-BLAST software, which we present here, incorporates the hybrid alignment algorithm for both homolog detection and model building, thus allowing the use of position-specific gap costs that were previously unfeasible. We show that Hybrid PSI-BLAST represents an improvement to the overall sensitivity and selectivity of homolog detection in comparison to NCBI PSI-BLAST.

2 METHODS

Hybrid PSI-BLAST has been developed in several steps. Here, we will focus on novel features, most notably the use of position-specific gap costs, and not repeat details already discussed in previous publications. However, we refer the interested reader to References (Li *et al.*, 2004, 2005; Yu and Hwa, 2001; Yu *et al.*, 2002) for more information on the hybrid alignment algorithm (Yu and Hwa, 2001; Yu *et al.*, 2002), its use in building position-specific substitution matrices (Li *et al.*, 2004) and the incorporation of information about suboptimal alignments into the model building (Li *et al.*, 2005). Since even the discussion of these newly implemented features is rather lengthy and technical, the reader most interested in the outcome of these is invited to skip directly to the Section 3.

The full Hybrid PSI-BLAST algorithm consists of essentially two major components—Hybrid Similarity Search and Hybrid Model Building. We thus discuss the details of each below.

Many of the implementation details depend on the choice of parameters such as cutoffs and other constants. We choose these parameters by

*To whom correspondence should be addressed.

optimizing the retrieval performance of the algorithm on a training dataset. In order to evaluate the performance of our remote homolog detection, we use SCOP (Andreeva *et al.*, 2004; Conte Lo *et al.*, 2002; Murzin *et al.*, 1995) that classifies sequences based on the relatedness of their protein structures. Specifically, we use for parameter optimization the ASTRAL40 subset of SCOP version 1.69 (7024 sequences) which contains sequences with less than 40% sequence identity—ensuring that evaluation is based on the ability to detect remote homologs. These parameters were then frozen and the performance of Hybrid PSI-BLAST was compared with that of NCBI PSI-BLAST (see Section 3) on an independent test dataset derived as the non-overlapping portion of ASTRAL40 version 1.75 (4974 sequences).

2.1 Hybrid similarity search

The goal of this phase is the identification of homologous sequences for inclusion in future model building phases. Since the gapless version of the hybrid and the classical Smith–Waterman algorithm (Smith and Waterman, 1981) are identical, Hybrid PSI-BLAST retains the word search heuristics of NCBI BLAST (Altschul *et al.*, 1990, 1997) for narrowing the list of candidate sequences. The full hybrid alignment algorithm is then used on these candidate sequences to determine whether or not they are homologs.

The hybrid alignment algorithm (Yu and Hwa, 2001) was so named because it contains features of both deterministic (Pearson and Lipman, 1988; Smith and Waterman, 1981) and probabilistic alignment algorithms (Durbin *et al.*, 1998; Hughey and Krogh, 1996; Loytynoja and Milinkovitch, 2003). In particular, this semi-probabilistic algorithm measures sequence homology according to the maximum log-likelihood score:

$$\Phi = \max_{\substack{1\leq i\leq M,\\ 1\leq j\leq N}} (\ln Z_{i,j}) \tag{1}$$

where $Z_{i,j}$ is the probabilistic likelihood score at sequence alignment positions i and j, and M and N represent the subject and query sequence lengths, respectively. Then, as in the case of deterministic alignment, the maximum score is used as the basis for evaluating sequence homology.

The significance of these scores can be assessed by using the same universal distribution derived for the case of gapless local alignment (Dembo *et al.*, 1994; Karlin and Altschul, 1990; Karlin and Dembo, 1992)

$$E(\Phi) = KMNe^{-\lambda\Phi} \tag{2}$$

where K and λ are the typical mode and tail parameters of the extremal distribution (Fisher and Tippett, 1928). However, in the case of hybrid alignment, we know a priori that $\lambda = 1$ for appropriately chosen scoring weights even in the presence of position-specific gap costs (Yu and Hwa, 2001)—a fact that we will exploit in this work. [The same feature is also known for profile Hidden Markov models (Eddy, 2008).] Thus, only the value of K needs to be evaluated in order to assess the statistical significance of our sequence hits.

2.2 Model building

There are a number of aspects to model building: choosing sequences to include in the model, weighing the contribution of each sequence appropriately and finally generating the position-specific scoring matrix (PSSM) that will act as the query in the next round.

2.2.1 Choosing sequences Sequences to be included in the model are chosen based on an E-value cutoff. In order to calculate E-values using Equation (2), the parameters K and λ have to be known. While $\lambda = 1$ is fixed, K depends on the scoring system. Thus, we recalculate the value of K in each round based on the new search model. More specifically, we use the island method (Altschul *et al.*, 2001) which determines the value of K from aligning random sequences of length 2000 amino acids to a model of length 2000 generated by concatenating several copies of the new search model. The length of 2000 is chosen in order to minimize edge effects. The number of such alignments is dynamically adjusted such that a statistical error of 5% on the estimated value of K is achieved. Similarly, the island method is used to estimate in every round the parameters H and β (Altschul and Gish, 1996; Altschul *et al.*, 2001), which are used to correct the length dependence of the E-values (Li *et al.*, 2004).

The actual choice of the inclusion threshold has to be empirically determined. A higher value will allow detection of more homologs but also increases the risk of model corruption by inclusion of false positives. After trying several cutoffs in steps of factors of 10, we settled on a cutoff of 0.0001. The fact that this is lower than the default cutoff of NCBI PSI-BLAST (0.005) is a reflection of the fact that the more detailed models with variable gap costs recognize homologs even at a smaller threshold.

While the database search itself is not much more computationally expensive using the hybrid algorithm than with the Smith–Waterman core, model building can become a major computational bottleneck if a very large number of homologs of the query are found. In order to avoid this bottleneck, we set an upper limit of 500 sequences that are included in the model building. If less than 500 homologs are found, all are used for model building. If more than 500 homologs are found, 500 of them are randomly sampled from the total list of homologs. Using this cutoff ensures that the computational load of Hybrid PSI-BLAST remains similar to that of NCBI PSI-BLAST. In practice, our version of Hybrid PSI-BLAST is about 2–3 times slower than NCBI PSI-BLAST on the same hardware.

Lastly, the ends of the sequences to be included in the model building have to be determined. It is important to focus only on highly scoring subsequences since otherwise residues are included in model building that are irrelevant for the protein family in question which can lead to the inclusion of non-homologs in the model in later rounds. Smith–Waterman alignment directly provides a beginning and end of its high scoring path. In hybrid alignment, the end is still well-defined but there is no beginning of the path. Instead of a hard beginning, there is a probability at each model position that the high scoring alignment has already started. For the purpose of model building, we define the beginning of the alignment to be the position where this probability exceeds 0.95 for the first time. In order to minimize contributions from neighboring domains to the model for the next round, we shorten the so determined sequence by three amino acids on each end.

2.2.2 Sequence weighting Since the sequence weighting algorithm in the original PSI-BLAST is based on the multiple alignment which in turn is a result of the Smith–Waterman algorithm, we had to develop an alternative sequence weighting scheme that is based on the quantities provided by the hybrid algorithm. In developing this weighting scheme, we strive to stay as close as possible to the intent of the NCBI PSI-BLAST weights, i.e. prevent sample biases in the database from dominating the query models (Altschul *et al.*, 1997).

The quantity calculated from the Smith–Waterman multiple alignment in the original PSI-BLAST is the weighted number of times an amino acid A occurs in column j of the multiple alignment, which yields the normalized frequency $q_{j,A}$ of amino acid A at model position j. This quantity is later used to determine the substitution score for amino acid A at position j. In the original PSI-BLAST, its initial value $q\prime_{j,A}$ is calculated by summing the weights of all sequences that contain the amino acid A at position j of the Smith–Waterman alignment with the query PSSM. Especially if the number of sequences in the multiple alignment is small, a matrix derived directly from the $q\prime_{j,A}$ may overfit the data. Thus, NCBI PSI-BLAST calculates the final frequencies using the pseudocount approach

$$q_{j,A} = \frac{\alpha_j q\prime_{j,A} + \beta_s g_{j,A}}{\alpha_j + \beta_s}. \tag{3}$$

Here, α_j is a measure of the number of sequences in the multiple alignment at position j which is calculated by the sequence weighting algorithm and β_s is the (substitution) pseudocount constant that has been chosen by the NCBI group (Schäffer *et al.*, 2001) for optimal retrieval performance. The pseudocounts $g_{j,A}$ themselves are calculated from the raw counts $q\prime_{j,A}$ by using the original non-position-specific matrix, such as BLOSUM62, for estimating the probabilities of seeing an amino acid substitution.

It is fairly natural to switch most of the model building process from Smith–Waterman alignments to hybrid alignments and take advantage of the information buried in suboptimal alignments. Instead of simply counting how often an amino acid A occurs in column j of the Smith–Waterman multiple alignment, we use the posterior probabilities $\Pr(A,j,X|\Omega)$ of finding amino acid A in sequence X at column j of the multiple alignment under the protein family model Ω. These probabilities can be readily obtained from the results of the forward and backward algorithms [the details of which can be found in (Yu *et al.*, 2002)] as

$$\Pr(A,j,X|\Omega)=\frac{\sum_{i|x_i=A}(f^M_{i,j}-1)\cdot b^M_{i,j}}{Z_{s_E,m_E}} \tag{4}$$

where $f^M_{i,j}$ and $b^M_{i,j}$ are the forward and backwards match weights at positions i and j in the subject and query sequences, respectively, and Z_{s_E,m_E} is the likelihood score satisfying Equation (1). Summing these over all sequences and normalizing yields

$$q''_{j,A}=\frac{\sum_X \Pr(A,j,X|\Omega)}{\sum_{X,a}\Pr(a,j,X|\Omega)}. \tag{5}$$

If the optimal alignment between sequence X and model Ω is dominant, its probability will be the leading term in $\Pr(A,j,X|\Omega)$. In this case, $\Pr(A,j,X|\Omega)$ has a single peak that is close to 1 for the amino acid A appearing in the optimal alignment and 0 for the other 19 amino acids. Thus, we simply 'count' the dominant amino acid in this position in the same way as done in the original PSI-BLAST. Otherwise, the distribution of $\Pr(A,j,X|\Omega)$ is flatter. A single sequence will contribute more than one amino acid to the multiple alignment at column j albeit with a reduced weight reflecting the uncertainty of the alignment at this position.

The goal of the weighting scheme is to calculate for every position j of the PSSM an effective number of sequences α_j to be used in the pseudocount Equation (3) and a weight $w_{X,j}$ for each sequence X. In the original PSI-BLAST, only sequences that have a match or mismatch at position j are used to calculate these quantities for position j, i.e. sequences with a gap at position j or the local alignment of which starts past position j or ends before position j do not contribute to these quantities. In addition, only the columns of the multiple alignment from an interval $[l_j,r_j]$ contribute in which *none* of the sequences that contribute have a gap or end.

In hybrid alignment, gaps and beginnings of alignments are not well defined but rather probabilistic in nature (the end of an alignment is, however, still well defined). Thus, we first introduce the probability

$$t_{X,j}\equiv\sum_a \Pr(a,j,X|\Omega) \tag{6}$$

that sequence X has *any* amino acid at position j in the probabilistic alignment (i.e. the complement of the probability that sequence X has a gap or not yet started or already ended at position j). We then introduce an arbitrary cutoff $\tau=0.95$ and treat sequences with $t_{X,j}>\tau$ as having an amino acid at position j and the others as not having an amino acid at position j. This allows us to define

$$\mathcal{X}_j\equiv\{X|t_{X,j}>\tau\} \tag{7}$$

as the set of sequences relevant for position j. The boundaries of the interval of positions relevant for position j then are given by

$$l_j \;\equiv\; \min\{k|\forall_{X\in\mathcal{X}_j}\forall_{m=k\ldots j}t_{X,m}>\tau\} \tag{8}$$

$$r_j \;\equiv\; \max\{k|\forall_{X\in\mathcal{X}_j}\forall_{m=j\ldots k}t_{X,m}>\tau\} \tag{9}$$

that is, they are the maximal choices of columns surrounding position j between which all of the sequences relevant for position j have at least probability τ for an amino acid.

In order to calculate the weight $w_{X,j}$ for sequence X at position j, we first calculate the 'frequencies'

$$f_{j,k,A}=\sum_{X\in\mathcal{X}_j}\Pr(A,k,X|\Omega) \tag{10}$$

of amino acid A at position k for all sequences that are relevant at position j. These are determined for all positions $k=l_j\ldots r_j$. Following the original PSI-BLAST weighting scheme, the contribution of amino acid A at position k is the inverse of the frequency $f_{j,k,A}$ of this amino acid times the inverse of the total number

$$F_{j,k}\equiv\#\{A|f_{j,k,A}>0\} \tag{11}$$

of amino acids that occur at all at position k. Since the $\Pr(A,k,X|\Omega)$ determine the probability of seeing amino acid A at position k in sequence X, the total weight $g_{j,k,X}$ contributed by position k to sequence X is

$$g_{j,k,X}\equiv\sum_A\frac{1}{f_{j,k,A}F_{j,k}}\Pr(A,k,X|\Omega) \tag{12}$$

(which we take to be zero if $f_{j,k,A}=0$ which implies $\Pr(A,k,X|\Omega)=0$). Since sequence X contributes to the alignment at position k with probability $t_{X,k}$, the total weight for sequence $X\in\mathcal{X}_j$ at position j is proportional to

$$w\prime_{X,j}\equiv\sum_{k=l_j}^{r_j}t_{X,k}g_{j,k,X}. \tag{13}$$

The unnormalized weights $w\prime_{X,j}$ of sequences $X\notin\mathcal{X}_j$ that are not relevant for position j are set to zero. Finally, the actual weights $w_{X,j}$ are calculated from the $w\prime_{X,j}$ by normalizing them at each position j.

In order to calculate the effective number of sequences α_j at position j, we first determine the diversity D_j of each position j. In the original PSI-BLAST, the diversity is the number of different amino acids that occur in this column of the multiple alignment. We replace this by the number of different amino acids that are the highest probability amino acid at position j within their sequence for all relevant sequences $X\in\mathcal{X}_j$. In order to calculate α_j, the diversities are averaged over all positions k between $\max\{j-5,l_j\}$ and $\min\{j+5,r_j\}$, i.e. over all positions from the relevant interval but no further away from j than five (the original PSI-BLAST weighting scheme uses the whole interval $[l_j,r_j]$ but we found that further restricting the number of positions over which to average gives better results). α_j is obtained by subtracting one from this average, since an average of one corresponds to all relevant sequences agreeing with the query sequence in which case no additional information can be retrieved from the identified homologs at this position. Thus, just like in the original PSI-BLAST weighting scheme, α_j is a number between zero and the alphabet size minus one.

Given the sequence weights $w_{X,j}$ and the effective numbers of sequences α_j, the raw 'frequencies' are determined as

$$q'_{j,A}=\frac{\sum_X w_{X,j}\Pr(A,j,X|\Omega)}{\sum_{X,a}w_{X,j}\Pr(a,j,X|\Omega)}. \tag{14}$$

Equation (3) is applied to determine the final frequencies $q_{j,A}$ which, in turn, are used to calculate the new PSSM.

2.2.3 PSSM There are two major parts to determining the PSSM: the amino acid substitution, i.e. match or mismatch, and the gap, i.e. insertion or deletion, weights. The substitution weights are dealt with in a similar way to NCBI's PSI-BLAST (Altschul *et al.*, 1997) with the inclusion of amino acid probabilities instead of a discrete alignment, and has already been described in detail by Li *et al.* (2005). Therefore, here we will focus on the inclusion of position-specific gap costs.

The main advantage of using the hybrid algorithm is that it allows position-specific gap costs in a natural way. In order to use position-specific gap costs, the PSSM has to be extended to include the substitution weight η_j and the gap weights $\mu_j^{I_1}$, $\mu_j^{I_2}$, $\mu_j^{D_1}$, $\mu_j^{D_2}$, ν_j^I and ν_j^D for every model position j. Correspondingly, the implementations of the (forward and backward) hybrid algorithm have to be modified to use these position-specific gap weights instead of the constant weights derived from the gap penalties supplied by the user.

During the first round of the iterative process, the additional parameters of the PSSM are initialized from the gap initiation cost δ and the gap extension cost ϵ supplied by the user. Specifically, they are calculated (Yu *et al.*, 2002) from the two parameters $\mu\equiv\exp[-\lambda_{ug}(\delta+\epsilon)]$ and $\nu\equiv\exp(-\lambda_{ug}\epsilon)$ where

λ_{ug} is the easily computable Gumbel parameter in the absence of gaps (Karlin and Altschul, 1990), as

$$\begin{aligned} \eta_j &= (1-\nu)^2/[(1+\mu-\nu)^2-\mu^2] \\ \mu_j^{I_1}=\mu_j^{D_1} &= [(1+\mu-\nu)^2-\mu^2]/(1-\nu) \qquad (15) \\ \mu_j^{I_2}=\mu_j^{D_2} &= \mu(1-\nu)/[(1+\mu-\nu)^2-\mu^2] \\ \nu_j^D=\nu_j^I &= \nu. \end{aligned}$$

where transition probabilities are represented by η_j, $\mu_j^{I_2}$ and $\mu_j^{D_2}$ for $M_j \rightarrow M_{j+1}$, $M_j \rightarrow I_j$ and $M_j \rightarrow D_{j+1}$, respectively. Since we disable the transitions between insertion state and deletion state, other transitions from node j to node $j+1$ are $I_j \rightarrow M_{j+1}$, $D_j \rightarrow M_{j+1}$ and $D_j \rightarrow D_j$ with the corresponding transition probabilities $\eta\mu_j^{I_1}$, $\eta\mu_j^{D_1}$ and ν_j^D, respectively. The transition $I_j \rightarrow I_j$ loops at node j. A detailed description of the hybrid alignment algorithm and of its parameters can be found in Li (2006). Note, that probability conservation requires in the framework of hybrid alignment that the weight for a gap initiation is split into the two pairs of contributions (μ^{I_1}, μ^{I_2}) and (μ^{D_1}, μ^{D_2}), respectively. The major question is how to choose these parameters in the subsequent iterations based on the subject sequences identified in the previous rounds.

A straightforward approach to answer this question is closely modeled after the determination of the substitution scores in the PSSM. From the results of the forward and backward algorithm, we can calculate for each sequence X and each position j the probability

$$\Pr(\eta,j,X|\Omega) \equiv \frac{\sum_i (f_{i,j}^M - 1)\cdot \mu_j^{D_2} \omega_j(x_i)\cdot b_{i+1,j+1}^M}{Z_{s_E,m_E}} \qquad (16)$$

that an alignment exists at position j and has a substitution at position j, the probability

$$\Pr(\mu^{I_2},j,X|\Omega) \equiv \frac{\sum_i (f_{i,j}^M - 1)\cdot \mu_j^{I_2}\cdot b_{i+1,j}^I}{Z_{s_E,m_E}} \qquad (17)$$

that an alignment exists at position j and has an insertion at position j, and the probability

$$\Pr(\mu^{D_2},j,X|\Omega) \equiv \frac{\sum_i (f_{i,j}^M - 1)\cdot \mu_j^{D_2}\cdot b_{i,j+1}^D}{Z_{s_E,m_E}} \qquad (18)$$

that an alignment exists at position j and is the beginning of a deletion. The new parameters η_j, $\mu_j^{I_2}$ and $\mu_j^{D_2}$ which encapsulate the relative weights of substituting, having an insertion or starting a deletion at position j should then be proportional to

$$q\prime_{j,W} \equiv \sum_X w_{X,j} \Pr(W,j,X|\Omega) \qquad (19)$$

for $W \in \{\eta, \mu^{I_2}, \mu^{D_2}\}$ which means we can assign

$$\eta\prime_j = \frac{q\prime_{j,\eta}}{q\prime_{j,\eta}+q\prime_{j,\mu^{I_2}}+q\prime_{j,\mu^{D_2}}}, \qquad (20)$$

$$\mu_j^{I_2\prime} = \frac{q\prime_{j,\mu^{I_2}}}{q\prime_{j,\eta}+q\prime_{j,\mu^{I_2}}+q\prime_{j,\mu^{D_2}}}, \qquad (21)$$

and

$$\mu_j^{D_2\prime} = \frac{q\prime_{j,\mu^{D_2}}}{q\prime_{j,\eta}+q\prime_{j,\mu^{I_2}}+q\prime_{j,\mu^{D_2}}}. \qquad (22)$$

The other weights are slightly more difficult to calculate. The deletion extension weight ν^D can be derived from the probabilities

$$\Pr(\nu^D,j,X|\Omega) = \frac{\sum_i f_{i,j}^D \cdot \nu_j^D \cdot b_{i,j+1}^D}{Z_{s_E,m_E}} \qquad (23)$$

that the alignment of sequence X has a deletion at position j and extends it and the probabilities

$$\Pr(\gamma^I,j,X|\Omega) = \frac{\sum_i f_{i,j}^D \cdot b_{i,j}^D}{Z_{s_E,m_E}} \qquad (24)$$

that the alignment of sequence X is in the deletion state before position j. Since ν^D represents the probability of extending a deletion conditional on being in a deletion, the new weight can be derived by dividing the total probabilities, i.e. as

$$\nu_j^{D\prime} = q\prime_{j,\nu^D}/q\prime_{j,\gamma^D} \qquad (25)$$

where $q\prime_{j,\nu^D}$ and $q\prime_{j,\gamma^D}$ are calculated according Equation (19) for $W \in \{\nu^D, \gamma^D\}$.

Calculating the insertion extension weight, ν^I is different from calculating the deletion extension weight, since a deletion contains only at most a single gap at model position j while in case of an insertion the complete insertion occurs in a single model position j. Thus, the essential quantity to extract from the results of the forward and backward algorithms is

$$L(j,X|\Omega) = \frac{\sum_i f_{i,j}^I \cdot b_{i,j}^I}{Z_{s_E,m_E}} \qquad (26)$$

which is the average number of insertion nodes visited at model position j or the average length of an insertion times the probability of having an insertion in the first place. This quantity can be averaged over the sequences to yield

$$L_j' \equiv \sum_X w_{X,j} L(j,X|\Omega). \qquad (27)$$

Since this quantity is the average length of an insertion times the probability that an insertion takes place, and $q\prime_{j,\mu^{I2}}$ calculated in Equation (19) is the probability that an insertion takes place, the average length of an insertion *given* that an insertion takes place is $L_j'/q\prime_{j,\mu^{I2}}$. This average length has to be translated into a value of ν_j^I. To this end, we note that every insertion at position j visits the first insertion node. The probability to visit the second insertion node is ν_j^I, the probability to visit the third insertion node is $(\nu_j^I)^2$, etc. Thus, the length distribution of an insertion is geometric with average $1/(1-\nu_j^I)$. Equating this to the measured average length of an insertion finally yields

$$\nu_j^{I\prime} = (L_j' - q\prime_{j,\mu^{I2}})/L_j'. \qquad (28)$$

The remaining two weights μ^{D_1} and μ^{I_1} depend on the others through normalization conditions and will be discussed at the end of this subsection.

Just as with the substitution weights, the transition weights determined in this way may be overfitting the data if there is only a small number of sequences contributing at model position j. Thus, we subject the raw weights $W \in \{\eta\prime, \mu^{I_2\prime}, \mu^{D_2\prime}, \nu^{D\prime}, \nu^{I\prime}\}$ to a pseudocount correction:

$$W_j = \frac{\alpha_j W_j' + \beta_g W_0}{\alpha_j + \beta_g} \qquad (29)$$

where α_j is the number of sequences contributing to positon j as calculated by the weighting scheme just as in Equation (3), β_g is a user-defined pseudo-count constant which can a priori be different from the pseudo-count constant β_s for the substitution weights, and W_0 is the original value of the corresponding weight according to Equation (15). We find $\beta_g = 1$ to be a good choice of the pseudocount constant.

This choice of pseudocount constant is lower than the value of $\beta_s = 10$ for the substitutions. This means that the presence or absence of gaps at a position of the model could have a strong influence on the ability of finding homologs with gaps at novel positions. Instead of increasing the pseudocount constant, we take a more refined approach. Gaps tend to correspond to loops in the protein and thus tend to occur in model positions that show lower levels of conservation. Thus, the expectation of gaps should be informed by the level of conservation in the corresponding positions as already implemented long ago in CLUSTALW (Thompson *et al.*, 1994). We thus calculate the information content

$$H_j \equiv \sum_A q_{j,A} \log_2(20 q_{j,A}) \qquad (30)$$

of the substitution scores at each position j and average these over positions $j-2,\ldots,j+2$ when calculating gap weights at position j. If this average is bigger than the empirically determined threshold 0.9, there is a large amount of conservation and we increase the weight for not having a gap by replacing η_j by $(1+\eta_j)/2$. If on the other hand, the average information content is below the threshold, we cap the weight η_j for not having a gap at $(1+\eta_0)/2$ where

η_0 is the user provided value of η for the first round of searches according to Equation (15). In any case, the value of η_j is capped on the bottom at 0.001 to always allow some probability for not having a gap and to avoid numerical artifacts. Once η_j is changed, the corresponding change is applied equally to $\mu_j^{I_2}$ and $\mu_j^{D_2}$ in order to continue to fulfill the normalization condition that requires those three values to add up to one.

Lastly, to avoid numerical artifacts, the weights ν_j^D and ν_j^I are capped at a minimum of 0.0001 and a maximum of 0.9999. Then, the normalization conditions for the weights are used to calculate the two remaining weights

$$\mu_j^{I_1} = (1-\nu_j^I)/\eta_j \tag{31}$$

and

$$\mu_j^{D_1} = (1-\nu_j^D)/\eta_j. \tag{32}$$

3 RESULTS

Using the parameter values we identified using SCOP v1.69, we measure the performance of both NCBI and Hybrid PSI-BLAST by comparison against an independent dataset that derives from the ASTRAL40 subset of SCOP v1.75. Of the 10 569 sequences in the ASTRAL40 subset of SCOP v1.75, 4974 are not contained in the ASTRAL40 subset of SCOP v1.69. We use the new sequences as an independent set of queries for testing the performance of Hybrid and NCBI PSI-BLAST.

We use version 2.2.18 of NCBI PSI-BLAST and Hybrid PSI-BLAST as described above to search the NCBI nr database for homologs of the query sequences in ASTRAL SCOP v1.75 that are not in ASTRAL SCOP v1.69 and save the resulting models after each round. These models are then used to search the ASTRAL SCOP v1.75 database and we ask how many proteins identified by NCBI and Hybrid PSI-BLAST are in agreement or disagreement by plotting the number of true versus false hits that occur, ordered by *E*-value. In order to determine which hits are true and which are false, we use the SUPERFAMILY evaluation ruleset for SCOP 1.69 benchmarks (Gough *et al.*, 2001; Madera *et al.*, 2004) which classifies each pair of sequences in SCOP as either homologs, non-homologs or undecidable. In general, two sequences in the same SCOP superfamily are considered as homologs, sequences in different folds are considered non-homologs and sequences in the same fold but different superfamily are considered undecidable. The authors of the evaluation ruleset also include a number of exceptions from this general classification scheme for common folds such as Rossmann folds and TIM barrels, as well as for individual sequences deemed to be misclassified by SCOP after manual inspection. Since a newer version of the ruleset is not available, we manually verified that none of the sequences covered by specific exceptions in the ruleset change classification between version 1.69 and 1.75. For the purpose of evaluating the algorithms, we ignore hits classified as undecidable.

The round by round results for NCBI PSI-BLAST are shown in Figure 1. From this plot, one can see the effects of the rather classic problem of model corruption in iterative approaches. As previously reported (Altschul *et al.*, 1997), NCBI PSI-BLAST performance initially improves in subsequent rounds, but then becomes worse past round 4 due to inclusion of a large number of non-homologs in the building of the query models.

Hybrid PSI-BLAST exhibits very little model corruption up to round 7, as shown in Figure 2. The robustness of Hybrid PSI-BLAST to model corruption in early rounds differentiates it from NCBI PSI-BLAST. Notice the first false hit in each subsequent round in Figures 1 and 2. The early false hits from the Hybrid PSI-BLAST iterations creep slowly to the right in comparison to NCBI PSI-BLAST.

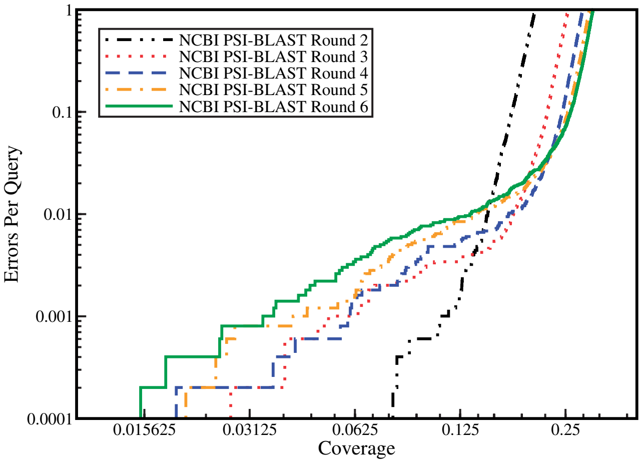

Fig. 1. NCBI PSI-BLAST round by round coverage versus errors per query. Coverage is plotted as the number of true hits identified by PSI-BLAST over the number of homologs in the database described in the main text. Errors per query are determined as the number of false hits over the total number of queries. Coverage versus errors per query are then plotted after sorting the hits by *E*-value. Here, we show the results for subsequent iterative rounds of NCBI PSI-BLAST from rounds 2 to 6. Round 1 consists of a pure BLAST search against the initial query. Round 2 is the first round in which a model is used for identifying homologs. We identify round 4 as providing the optimal trade-off between coverage and errors. This plot shows the sensitivity of the NCBI PSI-BLAST results to the number of iterations. The increasing number of early false hits in later rounds stems from model corruption—i.e. from the inclusion of non-homologs in the query models.

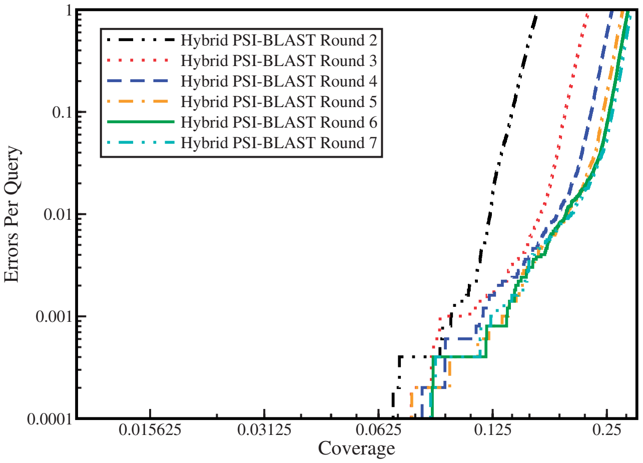

Fig. 2. Hybrid PSI-BLAST round by round coverage versus errors per query. The fraction of true hits in the database plotted versus the number of false hits per query for iterative rounds of Hybrid PSI-BLAST. Methods and definitions are the same as for those given for Figure 1. Here, the optimal performance is given by round 6. Note that the coverage of Hybrid PSI-BLAST, while still subject to the effects of model corruption, is less sensitive than that of NCBI PSI-BLAST to these effects.

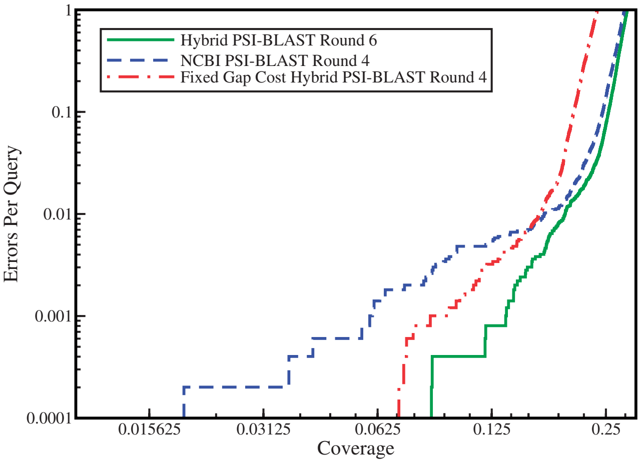

Fig. 3. Coverage versus errors per query for NCBI and Hybrid PSI-BLAST. The best iterative rounds of NCBI PSI-BLAST (round 4), Hybrid PSI-BLAST (round 6) and Hybrid PSI-BLAST without position-specific gap costs (round 4) are compared on the basis of the fraction of true hits found per fraction of false hits. Hybrid PSI-BLAST with position-specific gap costs performs better than both NCBI PSI-BLAST and fixed gap cost Hybrid PSI-BLAST, as can be seen from the fact that the Hybrid PSI-BLAST curve is downward and to the right, indicating that more homologs with fewer non-homologs were identified by Hybrid PSI-BLAST in comparison to NCBI PSI-BLAST. The difference between the fixed gap cost case and full Hybrid PSI-BLAST shows how much improvement is the result implementing position-specific gap costs.

Finally, we plot the best rounds for both algorithms and compare their overall performances. Figure 3 shows that Hybrid PSI-BLAST performs measurably better than NCBI PSI-BLAST. In order to determine if this improvement is due to the position-specific gap cost or due to the other changes in the algorithm, we repeat the evaluation of Hybrid PSI-BLAST keeping the gap weights constant at their initial value while keeping all other parameters the same. Figure 4 shows that in this case the model corruption effects are present but not as severe as for NCBI PSI-BLAST. Adding the performance of the best round of fixed gap cost Hybrid PSI-BLAST to Figure 3, we find that its performance is between that of NCBI and full Hybrid PSI-BLAST in the low coverage region. However, in the higher coverage region Hybrid PSI-BLAST without position-specific gap costs performs *worse* than NCBI PSI-BLAST. We thus conclude that incorporating position-specific gap costs indeed improves detection of remote homologies.

4 DISCUSSION

This work implements a version of PSI-BLAST that incorporates the semi-probabilistic hybrid alignment scheme. This alignment scheme allows for the use of position-specific gap costs and information from suboptimal alignments in the model building phase, resulting in overall more accurate homology searches than NCBI PSI-BLAST. The relevance of our result is that we have shown how to improve the very core of PSI-BLAST, i.e. the alignment engine, which was already backed by a long record of scientific scrutiny and sound theoretical foundations (Altschul *et al.*, 1990, 1997). Our performance improvements are orthogonal, and therefore potentially cumulative, to those brought about by more recent advances on other parts of PSI-BLAST such as the homolog detection and model building (Gonzalez and Pearson, 2010; Lee *et al.*, 2008).

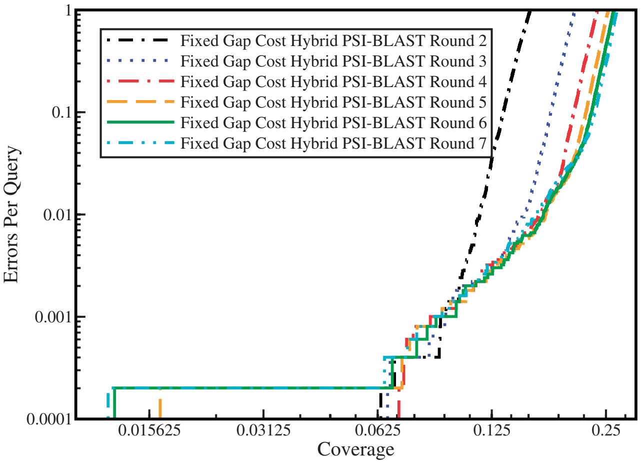

Fig. 4. Hybrid PSI-BLAST, without position-specific gap costs, round by round coverage versus errors per query. The fraction of true hits in the database plotted versus the number of false hits per query for iterative rounds of Hybrid PSI-BLAST. Methods and definitions are the same as for those given for Figure 1. Here, the optimal performance is given by round 4, two earlier rounds than for Hybrid PSI-BLAST with position-specific gap costs. Note that the coverage of the fixed gap cost Hybrid PSI-BLAST is more sensitive to model corruption than the full Hybrid PSI-BLAST algorithm, but less sensitive than NCBI PSI-BLAST.

ACKNOWLEDGEMENTS

The authors would like to thank Stephen Altschul for his helpful discussions.

Funding: This work was supported by the National Science Foundation (grant number 0317335). N.C. is thankful for partial support from the Institute for Genomic Biology Postdoctoral Fellows Program and Department of Energy (grant number DOE-2005-05818). This work was supported in part by an allocation of computing time from the Ohio Supercomputer Center.

Conflict of Interest: none declared.

REFERENCES

Altschul,S.F. and Gish,W. (1996) Local alignment statistics. *Methods Enzymol.*, **266**, 460–480.

Altschul,S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Altschul,S.F. *et al.* (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Altschul,S.F. *et al.* (2001) The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res.*, **29**, 351–361.

Andreeva,A. *et al.* (2004) SCOP database in 2004: refinements integrate structure and sequence family data. *Nucleic Acids Res.*, **32**, D226–D229.

Bateman,A. *et al.* (2002) The Pfam protein families database. *Nucleic Acids Res.*, **30**, 276–280.

Conte Lo,L. *et al.* (2002) Scop database in 2002: refinements accomodate structural genomics. *Nucleic Acids Res.*, **30**, 264–267.

Dembo,A. *et al.* (1994) Limit distribution of maximal non-aligned two-sequence segmental score. *Ann. Probab.*, **22**, 2022–2039.

Dokholyan,N.V. and Shakhnovich,E.I. (2001) Understanding hierarchical protein evolution from first principles. *J. Mol. Biol.*, **312**, 289–307.

Durbin,R. *et al.* (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK.

Eddy,S. (1998) Profile hidden Markov models. *Bioinformatics*, **14**, 755.

Eddy,S. (2008) A probabilistic model of local sequence alignment that simplifies statistical significance estimation. *PLoS Comput. Biol.*, **4**, e1000069.

Fisher,R.A. and Tippett,L.H.C. (1928) Limiting forms of the frequency distribution of the largest or smallest member of a sample. *Math. Proc. Camb. Philol. Soc.*, **24**, 180–190.

Gonzalez,M.W. and Pearson,W.R. (2010) Homologous over-extension: a challenge for iterative similarity searches. *Nucleic Acids Res*., **38**, 2177–2189.

Gough,J. *et al.* (2001) Assignment of homology to genome sequences using a library of hidden markov models that represent all proteins of known structure. *J. Mol. Biol.*, **313**, 903–919.

Hughey,R. and Krogh,A. (1996) Hidden Markov models for sequence analysis: extension and analysis of the basic method. *Bioinformatics*, **12**, 95.

Jones,D.T. (1999) Protein secondary structure prediction based on position-specific scoring matrices. *J. Mol. Biol*., **292**, 195–202.

Karlin,S. and Altschul,S.F. (1990) Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc. Natl Acad. Sci. USA*, **87**, 2264–2268.

Karlin,S. and Dembo,A. (1992) Limit distributions of the maximal segmental score among Markov-dependent partial sums. *Adv. Appl. Prob.*, **24**, 113–140.

Krogh,A. *et al.* (1994) Hidden Markov Models in Computational Biology:: applications to protein modeling. *J. Mol. Biol.*, **235**, 1501–1531.

Lee,M.M. *et al.* (2008) Simple is beautiful: a straightforward approach to improve the delineation of true and false positives in PSI-BLAST searches. *Bioinformatics*, **24**, 1339–1343.

Li,Y. *et al.* (2004) Using hybrid alignment for iterative sequence database searches. *CCPE*, **9**, 841–853.

Li,Y. *et al.* (2005) Suboptimal alignments improve the detection of weak homologs in sequence database searches. In *Proceedings of 5th International Conference on Bioinformatics and Bioengineering (BIBE)*, pp. 153–160.

Li,Y. (2006) Searching for Remotely Homologous Sequences in Protein Databases with Hybrid PSI-blast. PhD Thesis, The Ohio State University, Columbus, Ohio.

Loytynoja,A. and Milinkovitch,M.C. (2003) A hidden Markov model for progressive multiple alignment. *Bioinformatics*, **19**, 1505–1513.

Madera,M. *et al.* (2004) The SUPERFAMILY database in 2004: additions and improvements. *Nucleic Acids Res.*, **32**, D235–D239.

Murzin,A.G. *et al.* (1995) SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, **247**, 536–540.

Pearson,W.R. and Lipman,D.J. (1988) Improved tools for biological sequence comparison. *Proc. Natl Acad. Sci. USA*, **85**, 2444–2448.

Rondon,M.R. *et al.* (2000) Cloning the soil metagenome: a strategy for accessing the genetic and functional diversity of uncultured microorganisms. *Appl. Environ. Microbiol*., **66**, 2541–2547.

Schäffer,A.A. *et al.* (2001) Improving the accuracy of psi-blast protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res.*, **29**, 2994–3005.

Schwede,T. *et al.* (2003) SWISS-MODEL: an automated protein homology-modeling server. *Nucleic Acids Res.*, **31**, 3381–3385.

Smith,S.F. and Waterman,M.S. (1981) Comparison of biosequences. *Adv. Appl. Math.*, **2**, 482–489.

Thompson,J.D. *et al.* (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.

Yu,Y.K. and Hwa,T. (2001) Statistical significance of probabilistic sequence alignment and related local hidden markov models. *J. Comput. Biol.*, **8**, 249–282.

Yu,Y.K. *et al.* (2002) Hybrid alignment: high performance with universal statistics. *Bioinformatics*, **18**, 864–872.

Zhou,H. and Zhou,Y. (2005) Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. *Proteins*, **58**, 321–328.