ISSN 0378-1119
Volume 422, Issues 1–2, 1 October 2008
ELSEVIER
GENE
An International Journal of Functional and Evolutionary Genomics
Special Issue
Physical and Chemical Foundations of Bioinformatics Methods
Markus Porto, H. Eduardo Roman and Michele Vendruscolo (Editors)

Gene 422 (2008) 41–46

Contents lists available at ScienceDirect

Gene

journal homepage: www.elsevier.com/locate/gene

BCL::Align—Sequence alignment and fold recognition with a custom scoring function online

Elizabeth Dong, Jarrod Smith, Sten Heinze, Nathan Alexander, Jens Meiler [*]

Department of Chemistry and Center for Structural Biology, Vanderbilt University, Nashville, TN, USA

ARTICLE INFO

Article history:
Available online 7 June 2008

Received by: A. Bernardi

Keywords:
Dynamic programming
Fold recognition
Multiple sequence alignment
Needleman–Wunsch algorithm
Parametric sequence alignment

ABSTRACT

BCL::Align is a multiple sequence alignment tool that utilizes the dynamic programming method in combination with a customizable scoring function for sequence alignment and fold recognition. The scoring function is a weighted sum of the traditional PAM and BLOSUM scoring matrices, position-specific scoring matrices output by PSI-BLAST, secondary structure predicted by a variety of methods, chemical properties, and gap penalties. By adjusting the weights, the method can be tailored for fold recognition or sequence alignment tasks at different levels of sequence identity. A Monte Carlo algorithm was used to determine optimized weight sets for sequence alignment and fold recognition that most accurately reproduced the SABmark reference alignment test set. In an evaluation of sequence alignment performance, BCL::Align ranked best in alignment accuracy (Cline score of 22.90 for sequences in the Twilight Zone) when compared with Align-m, ClustalW, T-Coffee, and MUSCLE. ROC curve analysis indicates BCL::Align's ability to correctly recognize protein folds with over 80% accuracy. The flexibility of the program allows it to be optimized for specific classes of proteins (e.g. membrane proteins) or fold families (e.g. TIM-barrel proteins). BCL::Align is free for academic use and available online at http://www.meilerlab.org/.

1. Introduction

Sequence alignment and fold recognition are key computational tools for predicting the evolutionary history of proteins and detecting structurally related proteins from their amino acid sequence. The importance of these methods continues to increase with the exponential growth of sequence databases driven by various genome projects (Benson et al., 2007; Mewes et al., 1999). With the help of these tools, relationships are being determined between newly discovered sequences and existing sequence databases (Bairoch and Apweiler, 1998; Benson et al., 2006) along with proteins of known structure collected in the protein data bank (Berman et al., 2000). While sequence similarity frequently accompanies structural similarity as well as evolutionary relation to a common ancestor (Phillips et al., 2000; Castillo-Davis et al., 2004), one major goal of these comparisons is the assignment of a function to newly discovered sequences.

Yet it is known that many structurally homologous proteins can have very low sequence identity (Rychlewski et al., 2000) and in these cases sequence alignment methods alone provide little information. Threading algorithms (Jones, 1999; Lindahl and Elofsson, 2000) and sequence-only methods (Karplus et al., 1998; Rychlewski et al., 2000) for fold recognition have been specifically developed to predict structural similarity. However, the accuracy of most sequence alignment methods as well as the reliability of fold recognition methods is greatly diminished when comparing sequences in the so-called “Twilight Zone” with less than 25% sequence identity (Rost and Sander, 1993; Thompson et al., 1999).

Approaches to improve the accuracy of automatic sequence alignments start with the introduction of common substitution matrices such as PAM (Dayhoff et al., 1978) or BLOSUM (Henikoff and Henikoff, 1992). The progressive algorithm (Feng and Doolittle, 1987; Hogeweg and Hesper, 1984) implemented in MUSCLE (Edgar, 2004) uses probabilities derived from the PAM 240 matrix and position-specific gap penalties with iterative score refinement. ClustalW (Thompson et al., 1994) also uses a progressive alignment method and improves its accuracy by weighting sequences, customizing substitution matrix usage and changing gap penalties depending on the surrounding residues. Align-m (Van Walle et al., 2004) uses a non-progressive local approach to guide a global alignment. T-Coffee (Notredame et al., 2000) combines information from global and local sequence alignments to determine an optimized alignment. However, BLAST (Altschul et al., 1990) and PSI-BLAST (Altschul et al., 1997) continue to dominate the field of sequence alignment tools with their rapid word-based algorithm and the iterative search using position-specific score matrices.

While there is some overlap between the tools used for sequence alignment and fold recognition, there is significant emphasis on secondary structure prediction in fold recognition methods. Recent sequence-based methods (Lindahl and Elofsson, 2000; Rychlewski

[*] Corresponding author. 465 21st Ave South, BIOSCI/MRBIII, Room 5144B, Nashville, TN 37232-8725, USA. Tel.: +1 615 936 5662; fax: +1 612 936 2211.
E-mail address: jens.meiler@vanderbilt.edu (J. Meiler).

0378-1119/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.gene.2008.06.006

et al., 2000) include predicted structural information when generating the sequence-structure alignment. ORFeus (Ginalski et al., 2003b) uses a scoring matrix based on the PSI-BLAST profile and secondary structure prediction from PSIPRED (Jones, 1999). Threading-based algorithms like THREADER (Jones, 1999) evaluate template-based models of the target sequence using residue contact and hydrophobicity scores in a double dynamic programming algorithm. K*sync (Chivian and Baker, 2006) is a recent hybrid of both approaches that uses various weight sets to create an ensemble of sequence–sequence alignments. Based on this ensemble a library of models is created from which the optimal model is selected by tertiary structure analysis and energy prediction.

It was shown that fold recognition can be improved by incorporating the output of several primary fold recognition approaches in a secondary approach. Such meta-servers work by forming the consensus of several primary methods using either artificial neural networks (P-Cons) (Lundstrom et al., 2001) or more straight-forward structure comparison tools (3D-Jury) (Ginalski et al., 2003a).

With the growing number of sequence analysis and fold recognition tools being developed, it became clear that different scoring schemes can perform quite differently depending on the protein class, sequence identity level, or type of problem (fold recognition vs. sequence alignment). In turn, researchers often needs to invoke multiple tools to accomplish these tasks and it is difficult to determine which method produces the most accurate result given a particular scenario.

In the present study we seek to address this shortcoming by introducing BCL::Align. The program gives the user maximum flexibility in tailoring the scoring function to fit the specific problem. The effective scoring function used by BCL::Align is a linear combination of various substitution matrices, position-specific scoring matrices, secondary structure predictions, chemical properties, and gap penalties. Here, the algorithms implemented in BCL::Align are described and optimized parameter sets for four typical tasks are presented (sequence alignment and fold recognition in the 0–25% and 25–50% sequence identity regime). Results for the SABmark benchmark database (Van Walle et al., 2005) are compared with other leading sequence alignment tools. The significance of the weights is discussed in terms of their importance for sequence alignment and fold recognition at different levels of sequence identity.

2. Materials and methods

2.1. Needleman and Wunsch algorithm is employed for generation of optimal pairwise sequence alignment

BCL::Align uses a standard dynamic programming algorithm (Needleman and Wunsch, 1970) to optimally align two sequences A and B of length m and n. In order to execute the alignment, a scoring scheme for matches as well as gaps needs to be provided (see Section 2.2). The dynamic programming algorithm will output the optimal score $S_{m,n}$ together with the alignment.

Dynamic programming solves optimization problems by dividing the problem into independent subproblems. Since the sequence alignment problem has optimal substructure, a subproblem can be defined as aligning prefixes of two sequences up to a point (i,j) with $0<i\leq m$ and $0<j\leq n$. To find the alignment with the highest score $S_{m,n}$, a two-dimensional matrix with the dimensions m and n is filled at each position (i,j) with the best score $S_{i,j}$ of these prefix sequences ("matrix filling"). The optimal score $S_{i,j}$ builds upon the best score computed so far. The second part of the algorithm—so-called "trace back"—starts at the lower right corner of the matrix which now contains the best possible score $S_{m,n}$. It traces back step-by-step the pathway through the matrix that lead to this optimal score, thereby generating the optimal alignment of the two sequences.

2.2. Setup of parametric scoring function as a sum of weighted Z-scores

The scoring function of BCL::Align is a weighted sum of multiple scoring schemes that have been successfully used in prior sequence alignment and fold recognition approaches (discussed in Section 2.4). The user can choose the individual weight of each scheme and BCL::Align will recalibrate them to add up to 100%.

Raw scores obtained from each of the different scoring schemes are not directly comparable. Therefore all scores are first translated into Z-scores. For every scoring scheme, a random distribution was created by computing the score S for 10^6 arbitrarily chosen pairs of amino acids out of a representative database consisting of 1800 protein sequences. This database was created by culling the PDB (Berman et al., 2000) for sequences with less than 25% sequence identity (Wang and Dunback, 2003). For each of the different scores an average S_{av} and a standard deviation S_{sd} was derived (see Table 1) which are used within BCL::Align to rescale all scores into Z-scores with $Z=(S-S_{av})/S_{sd}$.

Therefore, positive scores larger than 1 indicate that two positions align with a score that is least one standard deviation above the average. Since the total score is a sum of weighted Z-scores, this statement holds not only for the individual scores but also for the total score, making all scores obtained with BCL::Align directly comparable even if the composition of the scoring function was altered.

2.3. Use of the affine gap penalty is essential for alignment of distant sequence homologs

The affine gap penalty approach (Barton and Sternberg, 1987) improves sequence alignment by customizing gap penalties to the sequence, making them length- and location-dependent. BCL::Align distinguishes gap open penalties P_{open} from gap extension penalties $P_{extension}$. It also distinguishes boundary gaps at the beginning or end of an alignment P^{B} from enclosed gaps P^{E}. In turn, a total of four gap penalties are defined that can be chosen by the user. The total penalty for a gap is computed using $P=P_{open}+\text{length}\times P_{extension}$.

2.4. Scoring function components were chosen from successful sequence alignment benchmarks and can be easily extended

Table 1 lists the parameter options open to the user. While substitution matrices of various sequence identity are available, the PAM250 (Dayhoff et al., 1978) and BLOSUM45 (Henikoff and Henikoff,

Table 1
Adjustable parameters and gap penalties

Description	Parameters	S_{av}[a]	S_{sd}[b]
Amino acid identity	Identity		
Substitution matrices	PAM 100, 120, 160, 250 (Dayhoff et al., 1978)	−0.0824	0.2498
	BLOSUM 90, 80, 62, 45 (Henikoff and Henikoff, 1992)	−0.0821	0.2273
Position-specific scoring matrix	BLAST profile (Altschul et al., 1997)	−0.0072	0.0881
Secondary structure predictions	PSIPRED (Jones, 1999)	−0.1431	0.4728
	JUFO (Meiler and Baker, 2003)	−0.0388	0.2451
	SAM (Karplus et al., 1998; Hughey and Korgh, 1996)	−0.0056	0.2076
Chemical properties	Steric parameter	−1.1514	0.8981
	Polarizability	−0.1061	0.0814
	Volume	−1.9938	1.5660
	Hydrophobicity	−1.0737	0.7871
	Isoelectric point	−1.6180	1.8058
Gap penalties	Open gap		
	Extension gap		
	Open boundary gap		
	Extension boundary gap		

[a] Average score for Z-score correction.
[b] Standard deviation for Z-score correction.

1992) matrices were used for sequence alignment because these matrices are most suitable for aligning sequences with low sequence identity. The logarithm of the probability of replacing amino acid *i* with *j* is used as the score.

The BLAST profile is iteratively built from members of the homologous family by scanning a sequence database (Altschul et al., 1997). In this work, the BLAST profile was determined by 3 PSI-BLAST iterations at an *E*-value cutoff of 0.001. The logarithm of the scalar product of the probability vectors for position *i* and *j* is used as the score. One advantage of using these parameters is that the scoring matrix obtained can be used directly for running PSIPRED and JUFO (see below).

The secondary structure predictions used in BCL::Align include PSIPRED (Jones, 1999), JUFO (Meiler and Baker, 2003) and SAM (Karplus et al., 1998; Hughey and Korgh, 1996). The logarithm of the scalar product of the 3-state (helix, strand, coil) probability vectors for position *i* and *j* is used as the score.

The chemical properties used include sterical parameters, polarizability, volume, hydrophobicity, and the isoelectric point which are also used as input for JUFO (Meiler and Baker, 2003). For scoring, the negative absolute difference for amino acids *i* and *j* is computed. After *Z*-score normalization, all five properties were combined with equal weights into a single score for weight optimization.

2.5. The SABmark benchmark database

For parameter optimization, we chose to use a subset of the 1.65 version of the SABmark reference alignment database (Van Walle et al., 2005), which is itself divided into two subsets. Sequences in the Superfamily subset have 25–50% sequence identity and are divided into test groups that represent different SCOP superfamilies. The Twilight Zone subset has sequences with 0–25% sequence identity and each test group represents a different SCOP fold.

SABmark also includes a second set of Twilight Zone and Superfamily subsets with the same sequences, plus the addition of up to the same number of false positive sequences. These false positives differ in fold from the true positives. They were selected from a BLAST search of the original sequences against a 70% identity subset of SCOP. The database covers the entire known fold space and each pairwise reference alignment is a consensus structural alignment provided by SOFI (Boutonnet et al., 1995) and CE (Shindeyalov and Bourne, 1998).

Because SABmark contained pairwise sequence alignments as well as fold information, we were able to use the benchmark to optimize the parameters for both the sequence alignment and fold recognition methods.

2.6. Optimizing the Cline score avoids over- and underprediction in sequence alignment

A total of eleven parameters and four gap penalties were optimized in our experiment (Table 1). For sequence alignment parameter and gap penalty optimization, we chose to maximize the Cline score (Cline et al., 2002) as a measure of alignment quality, finding in agreement with previous publications that maximizing the developer's score (f_d) alone leads to overprediction while maximizing modeler's score (f_m) leads to underprediction (Sauder et al., 2000; Edgar and Sjolander, 2004). Scores were calculated using the qscore program (Edgar, 2004, http://www.drive4.com/qscore/).

2.7. ROC curve analysis predicts accuracy of fold recognition

For fold alignment parameter optimization, we performed a receiver operating characteristic (ROC) curve analysis on the rate of correct versus incorrect fold assignment. A ROC curve plots the false positive rate against the true positive rate. Calculating the area underneath the ROC curve provides a measure of fold alignment accuracy, where an area of 50% would represent a program with no ability to recognize folds. The area underneath the ROC curve was maximized during parameter optimization.

Table 2
Training set on SABmark for parameter optimization

Problem	Sequence identity level [a]	Fraction of SABmark database used [b]		Score [c]
Sequence alignment	Twilight Zone	50%	873 of 1740 seq.	27
	Superfamilies	36%	1197 of 3280 seq.	49
Fold recognition	Twilight Zone	45%	1552 of 3458 seq.	82
	Superfamilies	22%	1460 of 6526 seq.	82

[a] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.

[b] The fraction of the SABmark database used for weight optimization is given as a percentage and in absolute sequences.

[c] Cline scores are reported for sequence alignment methods and the area under the ROC curve is reported for fold recognition methods. All scores are multiplied by 100. The maximum for both scores is 100.

2.8. Parameter and gap penalty optimization using a Monte Carlo algorithm

For both the sequence alignment and fold recognition methods, we performed two different optimizations, one with Twilight Zone sequences with low (0–25%) sequence identity and one with Superfamily sequences with intermediate (25–50%) sequence identity. For sequence alignment, the parameter and gap penalty optimization was performed on 50% of the Twilight Zone subset and 36% of the Superfamily subset. For fold recognition, 45% of the Twilight Zone subset and 22% of the Superfamily subset was used for the training set.

Using a Monte Carlo approach, we started the optimization with random values between 0 and 1 for the parameters and values between −2 and 0 for the gap penalties. For 100 Monte Carlo iterations, we adjusted the weights for the parameters and gap penalties by a random value between −0.2 and 0.2, maximizing the Cline score for sequence alignment and the area under the ROC curve for fold recognition. Fifteen rounds of this optimization procedure were carried out on each subset and weights from the top ten scoring rounds were averaged to determine the optimal weight set. The most favorable range for a particular weight is defined by average and standard deviation of the top ten scoring rounds of each trained subset.

2.9. Cross validation was used to avoid over-training

Since a subset of the SABmark database was used to determine the weight sets, we had to verify that the scores resulting from the parameter and gap penalty optimization were not affected by over-training. To do so, the scores for the trained and untrained subset were compared with each other. They were found to be within the standard deviation (Table 5), validating that the scores taken from the weight optimization can be directly compared with other leading methods.

2.10. Performance assessment

We assessed the sequence alignment performance of BCL::Align using the entire SABmark database. The average Cline scores for pairwise alignments in a group were calculated, and those scores were averaged to determine the final Cline score for each subset of SABmark: Twilight Zone, Superfamilies, Twilight Zone with False Positives, and Superfamilies with False Positives.

2.11. Implementation

The benchmarking and testing methods were written in C (using MPI for automated load balancing across a number of processors),

Table 3
Distribution of weights for parameters [a]

Problem	Sequence identity level [b]	PAM 250	BLOSUM 45	BLAST	PSIPRED	JUFO	SAM	Chemical properties [c]
Sequence alignment	Twilight Zone	0±0%	1±2%	36±5%	33±8%	16±11%	2±3%	11±3%
	Superfamilies	1±1%	2±2%	40±1%	35±5%	14±8%	1±2%	7±1%
Fold recognition	Twilight Zone	1±0%	19±6%	33±4%	30±4%	5±5%	5±5%	8±2%
	Superfamilies	0±1%	13±5%	18±5%	20±3%	18±3%	7±6%	24±4%

[a] Weight values, varying from 0 to 1.0, were normalized to calculate percentage of weight value out of 100%. Scores may not add to 100% due to rounding.
[b] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.
[c] Chemical properties include sterical parameters, polarizability, volume, hydrophobicity, and the isoelectric point.

with additional scripts written in Perl. Parameter optimization and performance assessment were performed on the PowerPC Linux cluster of the Vanderbilt University Advanced Computing Center for Research and Education (ACCRE).

3. Results and discussion

3.1. Optimal parameters and gap penalties

Details of the training set are given in Table 2, along with the average of the top ten scoring rounds of the Monte Carlo optimization. The optimized sequence alignment training set had an average Cline score of 27 for Twilight Zone sequences and 49 for Superfamily sequences. For fold recognition, the area underneath the ROC curve for the optimized training set scored an average of 82 for both subsets.

Tables 3 and 4 give the distribution of the sequence-identity dependent optimal weight sets for BCL::Align parameters and gap penalties for sequence alignment and fold recognition. The standard deviation on most weights is five percentage points or less, demonstrating the robust nature of the Monte Carlo optimization. However, we find that there is flexibility in the use of secondary structure elements for sequence alignment, particularly PSIPRED and JUFO. PSIPRED weights can vary up to 8 percentage points for the alignment of Twilight Zone sequences and 5 for Superfamily sequences. JUFO weights can vary up to 11 percentage points for Twilight Zone sequences and 8 for Superfamily sequences. The increase in standard deviation may be due to the various methods of secondary structure prediction compensating for each other in weight value, making their individual weights vary from one round to another.

For the gap penalties, we find that the same score is given by a consistent set of weights and the only range larger than 0.5 is found in the weight for the extension boundary gap for the alignment of sequences in the Twilight Zone subset at 0.6.

The relative weight of the parameters, expressed as percentages in Table 3, suggest that the BLAST profile and PSIPRED secondary structure information carry equal weight, within the standard deviation, for each of the four tasks. For sequence alignment, the BLAST profile has the highest average weight at 36% for the Twilight Zone subset and 40% for the Superfamily subset. This reiterates the power of position-specific scoring matrices created with PSI-BLAST as tools for sequence analysis. Amongst the secondary structure elements weights for alignment and fold recognition, we find that PSIPRED consistently carries the largest weight, with JUFO and SAM following behind. Only in the fold recognition of the Twilight Zone sequences do we find that JUFO and SAM carry equal weight at an average of 5%. For all other tasks, we find that JUFO outweighs SAM by over 10%. It is remarkable that the sum of the three secondary structure prediction weights is the largest contribution to the composite scoring function for all four benchmark cases.

The chemical properties of amino acids carry more weight in aligning sequences from the Twilight Zone at 11% compared to the 7% for Superfamily sequences. However, we find that the chemical properties are even more important in fold recognition, carrying 8% of the weight for the fold recognition of Twilight Zone sequences and 24% of the weight for the Superfamily subset. The relative importance of the PAM and BLOSUM substitution matrices is minimal in sequence alignment with weights below 2%, but we find that the BLOSUM matrix carries considerable weight in fold recognition at an average of 19% for Twilight Zone sequences and 13% for Superfamily sequences.

Large open gap and open boundary gap penalties were generally favored during parameter optimization of both the Twilight Zone and Superfamily subsets. The open gap penalty was −0.8 or more and the open boundary gap penalty was greater than −0.6 for all fold recognition and sequence alignment tasks. Generally, the extension gap and extension boundary gaps were penalized less, demonstrating the importance of the use of an affine gap penalty. We find that the extension boundary gap was penalized less than −0.3 for sequence alignment and fold recognition, as well as the extension gap for both sequence alignment tasks. However, for fold recognition there is a −1.3 penalty for Twilight Zone sequences and −1.4 for Superfamily sequences, indicating a particular emphasis on a penalty of the extension gap for fold recognition.

3.2. Cross-validation confirms absence of over-training

The scores for the trained and untrained subsets of SABmark for each of the four tasks are given in Table 5. In the Twilight Zone subset, the untrained subset had a Cline score of 24 whereas the trained subset had a score of 23. For the Superfamily subset, the untrained subset scored 51 while the trained subset had a score of 49. The scores

Table 4
Optimized weights for gap penalties

Problem	Sequence identity level [a]	Open gap	Extension gap	Open boundary gap	Extension boundary gap
Sequence	Twilight Zone	−1.4±0.3	−0.1±0.1	−0.7±0.4	−0.3±0.6
Alignment	Superfamilies	−1.9±0.1	−0.1±0.1	−0.9±0.2	0.0±0.1
Fold	Twilight Zone	−1.2±0.2	−1.3±0.2	−0.6±0.4	−0.2±0.1
recognition	Superfamilies	−0.8±0.4	−1.4±0.4	−1.7±0.3	−0.1±0.1

[a]The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.

Table 5
Scores on trained and untrained subsets of SABmark with optimal weight set

Problem	Sequence identity level [a]	Score for trained subset[b]	Score for test subset[b]
Sequence	Twilight Zone	23	24
alignment	Superfamilies	49	51
Fold	Twilight Zone	87	86
recognition	Superfamilies	88	86

[a] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.
[b] Cline scores are reported for sequence alignment methods and the area under the ROC curve is reported for fold recognition methods. All scores are multiplied by 100. The maximum for both scores is 100.

Table 6
Performance comparison of multiple sequence alignment programs on SABmark[a]

	Superfamilies[b]		Twilight Zone[b]	
	No FP[c]	With FP[c]	No FP[c]	With FP[c]
Align-m	44.75	41.53	15.93	13.72
ClustalW	47.60	47.82	18.57	17.98
T-Coffee	50.20	45.58	20.80	16.94
MUSCLE	44.52	40.38	15.45	12.44
BCL Align	**50.74**	**50.80**	**23.02**	**23.66**

[a] Cline scores are reported for each multiple sequence alignment program. The highest score in each subset is displayed in bold. Scores for all methods except BCL::Align are from Blackshields et al. (2006).
[b] The sequence identity level is 0–25% for the Twilight Zone subset and 25–50% for the Superfamily subset.
[c] Subsets include the addition of up to the same number of false positive sequences. False positives differ in fold from the true positives and were selected from a BLAST search of the original sequences against a 70% identity subset of SCOP.

for the untrained subsets of SABmark for sequence alignment are higher than those of the trained subset, providing evidence that the Monte Carlo optimization did not over-train the weight set and thus the method is not biased towards this particular subset. Although the scores of the untrained subset are lower than those of the trained subset for fold recognition, the difference is still within 2 percentage points. Nevertheless, BCL::Align would still benefit from future benchmarking tests on fold recognition benchmark databases such as the Lindahl Benchmark for fold-recognition sensitivity (Lindahl and Elofsson, 2000).

3.3. Comparison of sequence alignment methods

We compared the results of BCL::Align sequence alignment with Align-m (Van Walle et al., 2004), ClustalW (Thompson et al., 1994), T-Coffee (Notredame et al., 2000) and MUSCLE (Edgar, 2004) on the SABmark benchmark database using the Cline score. Scores for the methods listed above are from Blackshields et al., 2006. BCL::Align results on the entire SABmark benchmark database are shown in Table 6. In each subset, BCL::Align ranks the highest in alignment accuracy, demonstrating the superiority of BCL::Align's scoring function and the power of weight flexibility when compared to other programs that also use the dynamic programming algorithm (see Fig. 1). According to the data provided by Blackshields et al., ProbCons (Do et al., 2005) was the only program that consistently scored somewhat higher than BCL::Align. This is likely due to the fact that ProbCons does not employ dynamic programming but combines posterior-probabilities from pair-hidden Markov models (HMM) with a consistency-based method to determine scoring matrices.

3.4. Performance in fold recognition

There is not a universal score for measuring fold recognition accuracy. To determine the fold recognition accuracy of BCL::Align on the SABmark benchmark database subsets that included false positives, ROC curve analysis was performed. We find that BCL::Align has a strong performance, predicting the correct structure with 86% accuracy for the Superfamily subset and 83% accuracy for the Twilight Zone subset (see Fig. 2). However, the limiting factor for BCL::Align's ability to perform fold recognition is in the length of time it takes for the program to scan large databases in search of match fold, family and superfamily. Future improvements to increase the speed of BCL::Align using a word-based algorithm will allow for a more comprehensive study of the program's ability to perform fold recognition.

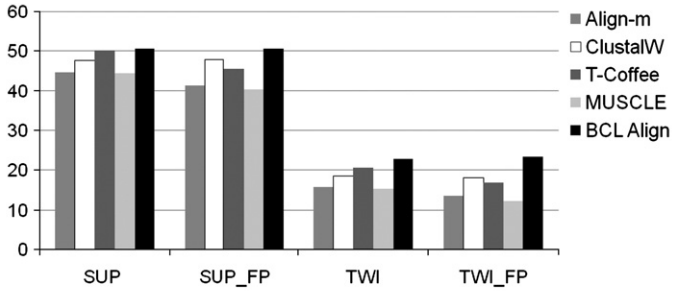

Fig. 1. Performance comparison of multiple sequence alignment programs on SABmark. Cline scores are reported for each multiple alignment program. Scores for all methods except BCL::Align are from Blackshields et al. (2006).

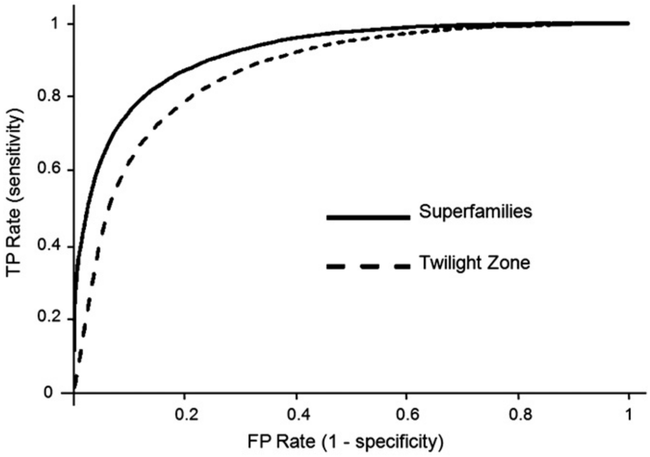

Fig. 2. ROC curve analysis of fold recognition on SABmark.

3.5. Conclusions

Sequence alignment and fold recognition at varying levels of sequence identity benefits from the use of customized weight sets because of the emphasis of different parameters for each situation. For the Superfamily subset, fold recognition puts an average of 12% more weight on chemical properties than sequence alignment. The BLOSUM45 substitution matrix carries over 10% more weight in fold recognition than sequence alignment. Of the secondary structure predictions, PSIPRED carries the most weight with 20–30% on average for all categories. JUFO follows behind with weights between 5 and 18%, and SAM has minimal involvement at less than 10% weight in all categories. In all cases, however, large weights for the BLAST profile and affine gap penalties provide optimal alignment and fold recognition. Using its optimal customized weight set, BCL::Align performed better than other dynamic-programming based methods with the highest rank in sequence alignment accuracy. With the future implementation of a faster word-based algorithm and the incorporation of HMM, we expect BCL::Align to have the efficiency to quickly align multiple sequences at once and perform fold recognition over large databases of protein structures. BCL::Align is available on an online web server at http://www.meilerlab.org/.

Acknowledgement

J.M. is supported by grant R01-GM080403 from the National Institute of General Medical Sciences. N.A. is supported by grant National Institutes of Health T32 GM08320, the Molecular Biophysics Training Grant at Vanderbilt University.

References

Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J., 1990. Basic local alignment search tool. J. Mol. Biol. 215, 403–410.

Altschul, S.F., Madden, T.L., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W., Lipman, D.J., 1997. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res. 25, 3389–3402.

Bairoch, A., Apweiler, R., 1998. The SWISS-PROT protein sequence data bank and its supplement TrEMBL in 1998. Nucleic Acids Res. 26, 38–42.

Barton, G.J., Sternberg, M.J., 1987. Evaluation and improvements in the automatic alignment of protein sequences. Protein Eng. 1, 89–94.

Benson, D.A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Wheeler, D.L., 2006. GenBank. Nucleic Acids Res. 34, D16–D20.

Benson, D.A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J., Wheeler, D.L., 2007. GenBank. Nucleic Acids Res. 35, D21–D25.

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne, P.E., 2000. The protein data bank. Nucleic Acids Res. 28, 235–242.

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

Boutonnet, N.S., Rooman, M.J., Ochagavia, M.E., Richelle, J., Wodak, S.J., 1995. Optimal protein structure alignments by multiple linkage clustering: application to distantly related proteins. Protein Eng. 8, 647–662.

Castillo-Davis, C.I., Kondrashov, F.A., Hartl, D.L., Kulathinal, R.J., 2004. The functional genomic distribution of protein divergence in two animal phyla: coevolution, genomic conflict, and constraint. Genome Res. 14, 802–811.

Chivian, D., Baker, D., 2006. Homology modeling using parametric alignment ensemble generation with consensus and energy-based model selection. Nucleic Acids Res. 34, e112.

Cline, M., Hughey, R., Karplus, K., 2002. Predicting reliable regions in protein sequence alignments. Bioinformatics 18, 306–314.

Dayhoff, M.O., Schwartz, R.M., Orcutt, B.C., 1978. A model of evolutionary change in proteins. In: Dayhoff, M.O. (Ed.), Atlas of Protein Sequence and Structure. National Biomedical Research Foundation, Washington, D.C., pp. 345–352.

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

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

Edgar, R.C., Sjolander, K., 2004. A comparison of scoring functions for protein sequence profile alignment. Bioinformatics 20, 1301–1308.

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

Ginalski, K., Elofsson, A., Fischer, D., Rychlewski, L., 2003a. 3D-Jury: a simple approach to improve protein structure predictions. Bioinformatics 19, 1015–1018.

Ginalski, K., Pas, J., Wyrwicz, L.S., von Grotthuss, M., Bujnicki, J.M., Rychlewski, L., 2003b. ORFeus: detection of distant homology using sequence profiles and predicted secondary structure. Nucleic Acids Res. 31, 3804–3907.

Henikoff, S., Henikoff, J.G., 1992. Amino acid substitution matrices from protein blocks. Proc. Natl. Acad. Sci. U.S.A. 89, 10915–10919.

Hogeweg, P., Hesper, B., 1984. The alignment of sets of sequences and the construction of phyletic trees: an integrated method. J. Mol. Evol. 20, 175–186.

Hughey, R., Krogh, A., 1996. Hidden Markov models for sequence analysis: extension and analysis of the basic method. Comput. Appl. Biosci. 12, 95–107.

Jones, D.T., 1999. GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. J. Mol. Biol. 287, 797–815.

Karplus, K., Barrett, C., Hughey, R., 1998. Hidden Markov models for detecting remote protein homologies. Bioinformatics 14, 846–856.

Lindahl, E., Elofsson, A., 2000. Identification of related proteins on family, superfamily and fold level. J. Mol. Biol. 295, 613–625.

Lundstrom, J., Rychlewski, L., Bujnicki, J., Elofsson, A., 2001. Pcons: a neural-network-based consensus predictor that improves fold recognition. Protein Sci. 10, 2354–2362.

Meiler, J., Baker, D., 2003. Coupled prediction of protein secondary and tertiary structure. Proc. Natl. Acad. Sci. U. S. A. 100, 12105–12110.

Mewes, H.W., Heumann, K., Kaps, A., Mayer, K., Pfeiffer, F., Stocker, S., Frishman, D., 1999. MIPS: a database for genomes and protein sequences. Nucleic Acids Res. 27, 44–48.

Needleman, S.B., 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.

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

Phillips, A., Janies, D., Wheeler, W., 2000. Multiple sequence alignment in phylogenetic analysis. Mol. Phylogenet. Evol. 16, 317–330.

Rost, B., Sander, C., 1993. Improved prediction of protein secondary structure by use of sequence profiles and neural networks. Proc. Natl. Acad. Sci. U. S. A. 90, 7558–7562.

Rychlewski, L., Jaroszewski, L., Li, W., Godzik, A., 2000. Comparison of sequence profiles. Strategies for structural predictions using sequence information. Protein Sci. 9, 232–241.

Sauder, J.M., Arthur, J.W., Dunbrack Jr., R.L., 2000. Large-scale comparison of protein sequence alignment algorithms with structure alignments. Proteins 40, 6–22.

Shindyalov, I.N., Bourne, P.E., 1998. Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Eng. 11, 739–747.

Thompson, J.D., Higgins, D.G., 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.

Thompson, J.D., Plewniak, F., Poch, O., 1999. A comprehensive comparison of multiple sequence alignment programs. Nucleic Acids Res. 27, 2682–2690.

Van Walle, I., Lasters, I., Wyns, L., 2004. Align-m—a new algorithm for multiple alignment of highly divergent sequences. Bioinformatics 20, 1428–1435.

Van Walle, I., Lasters, I., Wyns, L., 2005. SABmark—a benchmark for sequence alignment that covers the entire known fold space. Bioinformatics 21, 1267–1268.

Wang, G., Dunbrack Jr., R.L., 2003. PISCES: a protein sequence culling server. Bioinformatics 19, 1589–1591.