PROTEINS: Structure, Function, and Bioinformatics 64:960–967 (2006)

STRUCTFAST: Protein Sequence Remote Homology Detection and Alignment Using Novel Dynamic Programming and Profile–Profile Scoring

Derek A. Debe,[1] Joseph F. Danzer,[1] William A. Goddard,[2] and Aleksandar Poleksic[3]*
[1]*Eidogen-Sertanty Inc., San Diego, California 92121*
[2]*Materials and Process Simulation Center, California Institute of Technology, Pasadena, California 91106*
[3]*Department of Computer Science, University of Northern Iowa, Cedar Falls, Iowa 50614*

ABSTRACT **STRUCTFAST is a novel profile—profile alignment algorithm capable of detecting weak similarities between protein sequences. The increased sensitivity and accuracy of the STRUCTFAST method are achieved through several unique features. First, the algorithm utilizes a novel dynamic programming engine capable of incorporating important information from a structural family directly into the alignment process. Second, the algorithm employs a rigorous analytical formula for profile–profile scoring to overcome the limitations of ad hoc scoring functions that require adjustable parameter training. Third, the algorithm employs Convergent Island Statistics (CIS) to compute the statistical significance of alignment scores independently for each pair of sequences. STRUCTFAST routinely produces alignments that meet or exceed the quality obtained by an expert human homology modeler, as evidenced by its performance in the latest CAFASP4 and CASP6 blind prediction benchmark experiments. Proteins 2006;64:960–967.**

Key words: protein structure; homology modeling; comparative modeling; alignment algorithms; alignment statistics

INTRODUCTION

There are many different techniques for comparing and aligning protein sequences. Over the years, dynamic programming algorithms have dominated the field of protein sequence comparison. The widely applied BLAST[1] and FASTA[2] algorithms utilize dynamic programming in conjunction with a sequence–sequence scoring function that evaluates the similarities between the amino acids of the query and template sequence. The well-known PSI-BLAST[3] algorithm is an example of a sequence–profile method that replaces the query sequence with a profile of sequences from the query protein family. PSI-BLAST iteratively collects sequences from a sequence database to build a position-specific scoring matrix (PSSM). The PSSM is then used to search the sequence database for new homologs, which are used to construct a new position specific score matrix. This process is repeated until no new sequences are found. The hidden Markov model-based approaches, such as SAMT02[4] or HMMER use an explicit probabilistic model (HMM) in place of a position specific score matrix.

The new generation of profile–profile alignment methods utilizes multiple sequence alignment profiles in place of the query and template sequence.[5,6] Profile–profile algorithms enjoy additional enhancements in sensitivity, often recognizing sequences that share less than 15% identity, as evidenced by large-scale benchmarking experiments such as LiveBench,[7] CAFASP,[8] and CASP.[9–11]

Despite the success of profile–profile approaches, many aspects of their development remain ad hoc. For example, most of the existing profile–profile algorithms fail to provide a rigorous, probabilistic treatment for the column–column matching (as opposed to sequence–profile methods such as PSI-BLAST and HMMER, where the residue-column scores are treated as log-odd scores). The lack of a rigorous framework is universally true for profile–profile methods that incorporate other terms in the scoring function, such as position specific gap penalties and secondary structure information. To work well across the protein universe, these methods implement various ad hoc adjustments when scoring pairs of profiles, such as the normalization of the score matrix and score shifts.[5,6,12] This naturally introduces a performance bias toward the test sets used during parameter adjustment.

Estimating alignment score significance in profile–profile methods is also a challenging task. A rigorous measure of protein sequence similarity should reflect the difference between the quality of the best alignment of the two sequences and the quality of the best alignment between random sequences of the same lengths and compositions. For ungapped local sequence–sequence alignments in the asymptotic limit of long sequences, it is well established that the alignment scores follow an extreme value distribution described by two parameters λ and K.[13,14] For profile-based algorithms, with or without gaps, it has been conjectured that the score distribution is still of the Gumbel form. PSI-BLAST rescales the position specific

*Correspondence to: Aleksandar Poleksic, Department of Computer Science, University of Northern Iowa, Cedar Falls, IA 50614. E-mail: poleksic@cs.uni.edu

Received 8 February 2006; Revised 15 March 2006; Accepted 15 March 2006

Published online 19 June 2006 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/prot.21049

scoring matrix (PSSM) so that the scale of the raw scores corresponds to that of the standard substitution matrix. The assumption is that the random PSI-BLAST alignment scores, obtained by using a rescaled PSSM, will follow the Extreme Value Distribution of gapped sequence–sequence alignment scores. Thus, PSI-BLAST is able to quickly and precisely estimate the score statistics, without further random simulations. Due to the increased algorithm complexity, score normalization in profile–profile methods is a much more challenging problem. This is particularly true for algorithms that incorporate various constraints into the alignment process, such as secondary structure or complex gap treatments. For this reason, some profile–profile alignment methods use *Z*-score statistics to measure the alignment score significance.[5,12] COMPASS[15] provides a nice, PSI-BLAST-like statistical approach to computing *E*-values. However, the applications of this technique are limited to the specific scoring system used in the COMPASS algorithm.

In this article we introduce STRUCTFAST (**ST**ructure **R**ealization **U**tilizing **C**ogent **T**ips **F**rom **A**ligned **S**tructural **T**emplates), a novel, fully automated, profile–profile database search algorithm. The query sequence profiles in STRUCTFAST are generated with a modified version of the PSI-BLAST algorithm. A database of profiles for template representatives from the Protein Data Bank (PDB) are generated in a similar manner, but are augmented with information from structure–structure alignments derived from the template protein's structural family. A query profile is then aligned and scored against the library of structural profile templates and the alignments are ranked by the significance of their scores, as determined by the previously published Convergent Island Statistics method.[16]

MATERIALS AND METHODS

Profile Construction

STRUCTFAST uses an internally modified version of PSI-BLAST to compare a protein sequence of interest against the NCBI's nonredundant *nr* database. After 10 PSI-BLAST iterations, the algorithm parses the checkpoint file to obtain the probabilities (target frequencies) p_i, where $i = 1,2, \ldots .20$ for the 20 different amino acids at each sequence position. In addition, our version of PSI-BLAST reports, for each sequence position, weighted frequencies r_k, where $k = 1,2 \ldots 20$ for the 20 different amino acids (these are called "observed residue frequencies f_k" in PSI-BLAST) as well as the mean number of different residue types (including the gap character) observed in a neighborhood of the profile column C.[3] This alignment variability measure, denoted by N_C, saturates at 21 and is used in PSI-BLAST to weight the contribution of the column's observed amino acid frequencies in estimating the probabilities of 20 amino acid residues in the profile column C. In STRUCTFAST r_k and N_C are used in a different manner, as explained below.

STRUCTFAST also takes advantage of PSI-BLAST's ability to take a multiple sequence alignment instead of a single sequence as input. For structural templates, the input to PSI-BLAST is a multiple alignment consisting of the PDB structures in the template's Dali structural alignment.[17] This is a standard technique often used to increase the sensitivity of the database search.

Profile–Profile Scoring Function

In PSI-BLAST, the score for aligning residue R to profile column C is equivalent to

$$s(C,R) = \log\frac{P_c(R)}{P_B(R)}, \quad (1)$$

where $P_C(R)$ is the probability of observing letter R in column C (e.g., its target frequency) and $P_B(R)$ is the overall probability of the residue R (background frequency). A physical analogy of this scoring scheme consists of throwing weighted 20-sided dice. $P_C(R)$ is the probability of rolling R on the die C, where the area of each of the sides on die C is proportional to the frequencies of each amino acid in the template profile. $P_B(R)$ is the probability of rolling R on the "background die" B, where the area of each of sides on this die are proportional to the background amino acid frequencies.

The same approach and analogy can be applied to profile–profile scoring.[15] For profile–profile scoring, instead of computing the probability of a single event (corresponding to a single residue R), we need to compute the probability of a series of events (corresponding to the second profile column C_2). Thus, in the context of profile–profile scoring, Equation (1) has the following form:

$$s(C_1,C_2) = \log\frac{P_{C_1}(C_2)}{P_B(C_2)} \quad (2)$$

A standard manipulation can be employed to make scoring function 2 symmetric with respect to C_1 and C_2:

$$\text{score}(C_1,C_2) = s(C_1,C_2) + s(C_2,C_1) \quad (3)$$

The COMPASS algorithm implements a variant of Equation (3) along with fixed gap penalties. In COMPASS, the contribution of each score $s(C_1,C_2)$ and $s(C_2,C_1)$ is weighted and the weights are optimized by running the algorithm on large validation sets.

The STRUCTFAST scoring scheme employs a direct, nonweighted calculation of Equation (3). STRUCTFAST scores depend not only on the similarity between the profile columns, but also on the "amount of confidence" (called "thickness" in the COMPASS algorithm) in both sequential profiles. In other words, even the scores between similar columns are not high unless there is "confidence" in the quality of both sequential profiles. This dependence of the scores on the quality of the profiles in STRUCTFAST establishes the need for an appropriate scaling of the gap penalties, and also renders the raw scores (or any alignment statistics based on a fixed background distribution) useless in ranking the significance of the database hits. In other words, the choice of the scoring function (3) mandates the need for (a) profile-pair specific gap penalties, and (b) profile-pair specific score statistics.

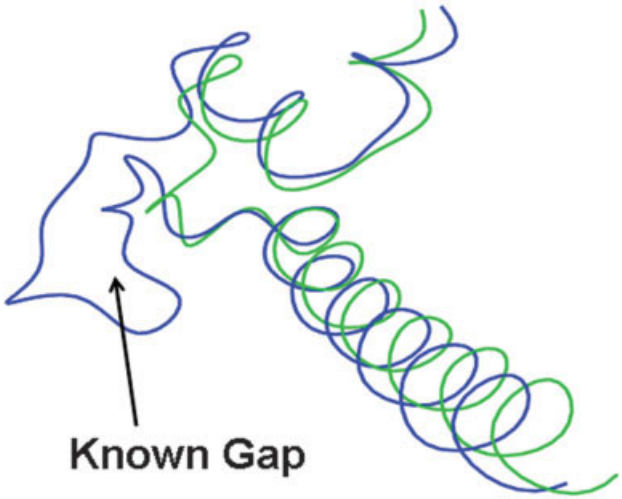

Fig. 1. Structural overlay of 1jnqA (blue) with 1lox (green). The significantly different loop lengths at the upper left of the overlay lead are recorded as a STRUCTFAST BRIDGE/BULGE gap.

TABLE I. The Structure Alignment Produced by the Program Dali[17] for the Protein Domains 1ovaA and 1by7A (the C-Terminus of the Alignment has been Truncated at Residue 189 of 1ovaA)

	1ovaA	1by7A
Aligned	1–63	1–63
Gap	64	
Aligned	65–68	64–67
Gap	69–78	
Aligned	79–91	68–80
Gap	92–97	
Aligned	98–189	81–172

The first 63 and the last 91 residues in the structures are aligned. Residues 69–78 in 1ovaA do not align to any residues in 1by7A, even though the structures are similar on both sides of the gap. Thus, with respect to 1by7A, 1ovaA has a 9-residue BULGE in this region. Conversely, with respect to 1ovaA, the structure 1by7A BRIDGES 9 residues in this region of 1ovaA.

STRUCTFAST's profile-pair specific gaps and score statistics are discussed in greater detail in the following sections.

An important question that requires further analysis is how to define an "outcome" (C_2). In STRUCTFAST, the number of times residue R_k is observed in an outcome is set to $n_k = N_C * r_k$, where r_k and N_C are the values described above. Because the experiment of rolling a die N times and getting n_k occurrences of outcome k is described by a multinomial,

$$P_{C_1}(C_2) = N! \prod_{k=1}^{20} \frac{p_k^{n_k}}{n_k!} \quad (4)$$

In Equation (4), the values $\{p_k\}^{20}_{k=1}$ are target frequencies for residues in the first column (C_1), $\{n_k\}^{20}_{k=1}$ are the effective residue counts in the second column (C_2), and $N = \sum_{k=1}^{20} n_k$.

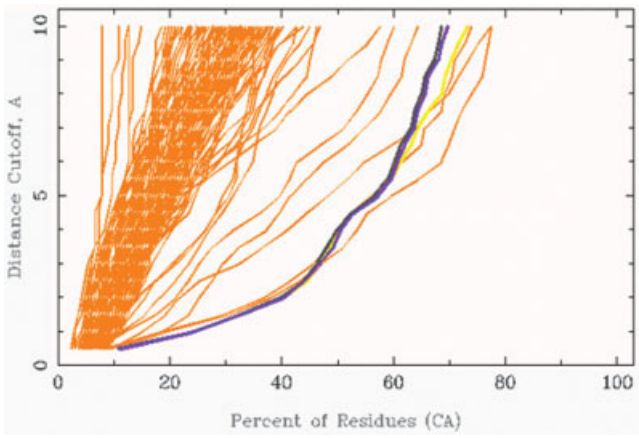

Fig. 2. The CASP6 Hubbard plot for Target T0197. STRUCTFAST-SFST is shown in black, STRUCTFAST-BNMX is shown in violet, and STRUCTFAST-EXPM is shown in yellow. The two orange lines that track the STRUCTFAST performance belong to the hand modeling groups Baker and GeneSilico-Group. This target tricked nearly everyone in the competition.

To compute $P_B(D)$, the target frequencies p_k for the background column (die) B are set to the background probabilities of 20 amino acids, such as Robinson and Robinson.[18]

Because of the rigorous analytical treatment of the profile–profile scores, the STRUCTFAST scoring function contains no parameters to optimize. An average STRUCTFAST score between two randomly chosen profiles is a small negative number, which implies that the algorithm stays in the local regime, that is, that the distribution of random alignment scores follows an Extreme Value Distribution.

Gap Treatment—STRUCTFAST Dynamic Programming

Incorporating structural information into the alignment process is known to improve the sensitivity of the alignment algorithm. For example, FUGUE[19] uses environment-specific substitution matrices and structure-dependent gap penalties derived from structure–structure alignments in the HOMSTRAD database.[20] The scoring function in ORFEUS[12] assigns higher scores to pairs of residues predicted to be in the same secondary structure state. The gap penalties in CLUSTALW[21] and MODELLER[22] are altered based on the environment of the particular gap, for example, whether or not the gap is located within a template secondary structure (high penalization) or loop region (mild penalization). One of the unique features of STRUCTFAST is a novel dynamic programming approach that incorporates gap information from a structural family directly into the alignment process.

STRUCTFAST's dynamic programming engine allows for two types of gaps. The first type of gaps are standard, affine gaps that can be introduced anywhere in the alignment. The gap penalties for these gaps are computed according to the standard formula:

$$P = O + (l - 1)E \quad (5)$$

TABLE II. A Partial List of the STRUCTFAST BRIDGE/BULGE Information for the Template 1ovaA

Aligned Protein	Gap Type	Start Residue in 1ovaA	Stop Residue in 1ovaA	No. of Aligned Residues
1ovaC	BRIDGE	341	354	1
1ovaB	BRIDGE	65	79	1
1azxI	BULGE	24	25	2
1azxI	BULGE	62	63	3
1azxI	BRIDGE	66	78	1
1azxI	BULGE	92	94	3
1azxI	BRIDGE	223	225	1
1azxI	BRIDGE	269	272	1
1azxI	BULGE	308	309	2
1azxI	BULGE	316	317	3
1azxI	BULGE	338	341	8
1azxI	BRIDGE	345	348	2
1azxI	BRIDGE	351	353	1
1by7A	BRIDGE	63	65	1
1by7A	BRIDGE	68	79	1
1by7A	BRIDGE	91	98	1
1by7A	BRIDGE	189	193	1
1by7A	BULGE	25	237	1
1by7A	BULGE	249	250	5
1by7A	BULGE	308	309	2
1by7A	BRIDGE	339	355	1

where l is the length of the gap and O and E are opening and extension penalties, respectively. The second types of gaps are special, BRIDGE/BULGE gaps. BRIDGE/BULGE gaps are possible only in certain parts of the alignments, namely those that correspond to the regions in the template that are inserted or deleted in the template's structural alignment with other PDB structures (Fig. 1).

In practice, a BRIDGE/BULGE list for every template is precalculated, after being harvested from an $N \times N$ structure alignment of every structure in the PDB. Table I shows a structural alignment of 1ovaA against 1by7A, and explains how the BRIDGE/BULGE gaps are derived. Table II shows a partial BRIDGE and BULGE list of information for the template protein 1ovaA.

BRIDGE/BULGE gaps are incorporated into the dynamic programming algorithm via the following recurrence formula:

$$S_{ij} = \max\{mch_{ij}, del_{ij}, ins_{ij}, bb_{ij}\}, \quad (6)$$

where

$$mch_{ij} = S_{i-j-1} + s_{ij},$$

$$del_{ij} = S_{i-p,j-1} + Agap(p-1), p > 1$$

$$ins_{ij} = S_{i-1,j-q} + Agap(q-1), q > 1$$

$$bb_{ij} = S_{i-p,j-q} + BBgap(j-q, j, p-1), p > 1 \text{ or } q > 1$$

In Equation (6), s_{ij} is the score for matching profile columns i and j, $Agap(k)$ is the affine gap penalty for the gap of length k, and $BBgap(a,b,l)$ is the gap penalty for the BRIDGE/BULGE gap of length l that spans the template positions a and b.

Profile-Pair Specific Gap Penalties

In the STRUCTFAST algorithm, the gap opening and extension penalties for a given pair of profiles are closely related to the distribution of profile–profile scores. For a pair of profiles P_1 and P_2, the gap opening penalty $O(P_1,P_2)$ and the gap extension penalty $E(P1, P_2)$ are computed as:

$$O(P_1,P_2) = \min(4*\text{stdev}(P_1,P_2), \text{max\_score}(P_1,P_2))$$

$$E(P_1,P_2) = O(P_1,P_2)/10 \quad (7)$$

where max_score(P_1,P_2) is the maximum column–column score and stdev(P_1,P_2) is the standard deviation of the distribution of column–column scores for P_1 and P_2. Because BRIDGE/BULGE gaps are observed in nature, the BRIDGE/BULGE opening and extension penalty is set to be 10 times lower than the penalty for an affine gap of the same length.

The parameters in the Equation (7) are not optimized. We simply followed some known approaches in defining the alignment scoring scheme. For example, in many search algorithms, the gap opening penalty is close to the maximum entry in the alignment score matrix. However, the nature of STRUCTFAST's scoring scheme establishes a need for placing an upper bound 4* stdev(P_1,P_2) for the gap opening penalty, as the maximum column–column score is sometimes too high.

Following the same idea, the gap opening/gap extension ratio (10) in the Equation (7) is very close to the parameter used in many state of the art algorithms for protein sequence alignment (e.g., BLAST).

More research is needed to see whether the optimization of the above two values would yield a noticeable improvement in STRUCTFAST's performance.

Note that the time complexity of the algorithm according to equation 6 is $O(n^3)$. To gain speed, STRUCTFAST uses a known technique to reduce the computational time for the standard part of the algorithm (without the BRIDGE/BULGE term) to $O(n^2)$. As BRIDGE/BULGE gaps are relatively sparse, for a majority of template sequences, the total execution time is dominated by the $O(n^2)$ term.

Alignment Score Significance

The score significance in the STRUCTFAST algorithm is estimated using Convergent Island Statistics (CIS).[16] The CIS algorithm builds upon the island statistics method[23] and is generally applicable to any class of algorithms that generates local alignments. The main idea behind our Convergent Island Statistics method is to recognize the lack of sequence similarity early in the shuffling process and thus save on the search time. In other words, for a given pair of sequential profiles, STRUCTFAST computes conservative estimates of the distribution parameters λ and K based on a small number of profile shuffles, and uses these values to decide whether to keep or discard any particular hit. Because any given sequence typically has a small number of significant hits in a representative, large database, the vast percentage of comparisons will be computed very efficiently. On the other hand, if the sequences are related (or show significant promise of being

TABLE III. Official Ranking of the Top Individual (Non-meta) Servers in the Fold Recognition Category at CAFASP4

Server	Total score
STRUCTFAST-SFST	701
STRUCTFAST-EXPM	692
STRUCTFAST-BNMX	678
Raptor	634
Bas_C	582
Mbam	581
Bas_B	574
Sp_3	561
—	—
PDB-BLAST	84
—	—
BLAST	28
—	—

Fold recognition targets roughly correspond to targets that do not have a good parent structure in the same superfamily. A total of 70 automated servers entered CAFASP4, and the complete table of results can be found at http://www.cs.bgu.ac.il/~dfischer/CAFASP4/frn1. The results of the well-known PDB-BLAST and BLAST algorithms are provided for reference.

TABLE IV. Official Ranking of Individual Servers in the Homology Modeling Category at CAFASP 4 (http://www.cs.bgu.ac.il/~dfischer/CAFASP4/hm1)

Server	Total score
STRUCTFAST-EXPM	2058
Inub	2017
STRUCTFAST-BNMX	2010
Shgu	2005
Spk2	1960
Shub	1996
STRUCTFAST-SFST	1999
Sp_3	1944
—	—
PDB-BLAST	1859
—	—
BLAST	1294

Homology modeling targets are those with a good parent structure in the same SCOP superfamily. The other well-performing autonomous servers include consensus methods Shub, Inub, and Shgu[25] and two variants of the SPARKS method.[26,27]

TABLE V. The Ranking of Servers According to Their Average Specificity at CAFASP4 (http://www.cs.bgu.ac.il/~dfischer/CAFASP4/specall)

Server	Average specificity
STRUCTFAST-EXPM	46.667
Shub	46.500
STRUCTFAST-SFST	45.833
STRUCTFAST-BNMX	45.667
Bas_C	45.667
Mbam	44.833
Bas_B	44.000
Sparks	43.833
—	—
PDB-BLAST	34.333
—	—
BLAST	21.500

Aside from the STRUCTFAST and the Shub[25] method, which also perform well in the sensitivity measures, other high specificity methods include servers from BioinfoBank in Poland, Bas_C, Bas_B and Mbam.[28]

related), CIS approaches the complexity of the brute force method of doing extensive random shuffles and precisely estimates the alignment statistics. The payoff of this rigorous statistical approach to score normalization will be illustrated later in Example 1.

Because the Convergence Island Statistics is the most time-consuming procedure in STRUCTFAST, the time needed to search a database of templates with our method mainly depends on the number of database sequences "similar" to the query sequence. Although we have not done an extensive analysis of the STRUCTFAST's time efficiency, our experience shows that the average time to search PDB with STRUCTFAST on a 2.66-GHz Pentium 4 machine is about 2 h. However, the time complexity distribution is spread out, resulting in the search time for some sequences approaching 10 h or more.

RESULTS AND DISCUSSION

Three different versions of the STRUCTFAST algorithm—SFST, BNMX, and EXPM—were entered as fully automated prediction servers in the recent CAFASP4 and CASP6 benchmark experiments. SFST is a straightforward implementation of the techniques described in this article, reporting an alignment to a single PDB template. BNMX and EXPM report all atom coordinates. The first set of alpha carbon coordinates in both BNMX and EXPM is derived from the best SFST alignment. Additional alpha carbon coordinates are derived from the second best (statistically significant) SFST alignment, but only if there is an overlap between aligned query residues in the first and the second alignment. This is a standard technique, frequently used in many comparative modeling methods such as MODELLER. The remaining backbone atoms are reconstructed from the alpha carbon coordinates.[24] The difference between BNMX and EXPM lies in the choice of the null model for protein sequence families. BNMX assumes a "flat" background model (the background probability of every amino acid is set to 0.05), whereas EXPM employs Robinson and Robinson background frequencies.[18]

We entered three different servers into the prediction experiments because of the wide variation in the metrics that are used to evaluate model quality. Some evaluation metrics tend to reward longer models, even if the additional residues are modeled less precisely. Other evaluation metrics will tend to penalize longer models if the additional residues are modeled less precisely. We expected that BNMX and EXPM would outperform SFST for evaluation metrics where longer models are rewarded, and that SFST would perform best under evaluation metrics where longer models are penalized.

TABLE VI. Rankings for the Top 30 Human Modelers and Servers in the Homology Modeling Category at CASP6, as Measured by the Sum of the Official Raw GDT Scores for the Easy and Hard Comparative Modeling Categories

Overall rank	Server rank	CASP6 Group Name	CM-EASY average GDT	CM-HARD average GDT	Total CM average GDT
1		Ginalski	1989.19	1173.03	3162.22
2		Skolnick-Zhang 2	1953.58	1138.49	3092.07
3		KOLINSKI-BUJNICKI	1935.24	1105.17	3040.41
4		GeneSilico-Group	1889.38	1102.23	2991.61
5		CHIMERA	1893.16	1051.56	2944.72
6	**1**	**STRUCTFAST-EXPM**	1887.84	1044.60	2932.44
7		SBC-Pmodeller5	1882.07	1043.79	2925.86
8	**2**	**ZHOUSPARKS2**	1883.86	1041.27	2925.13
9		FISCHER	1848.65	1075.18	2923.83
10		Jones-UCL	1879.56	1042.42	2921.98
11		SBC	1898.90	1020.71	2919.61
12		TOME	1895.54	1019.10	2914.64
13		Sternberg	1870.24	1043.89	2914.13
14		CBRC-3D	1881.94	1029.81	2911.75
15		CMM-CIT-NIH	1832.65	1076.22	2908.87
16		CAFASP-Consensus	1853.74	1051.45	2905.19
17		BAKER	1887.77	1012.84	2900.60
18	**3**	**zhousp3**	1861.71	1038.13	2899.84
19		SAM-T04-hand	1836.36	1050.62	2886.98
20	**4**	***ACE(Meta)***	1851.86	1032.12	2883.98
21		MCon	1876.49	991.80	2868.29
22		3D-JIGSAW	1851.69	1016.01	2867.70
23	**5**	**STRUCTFAST-BNMX**	1863.88	998.77	2862.65
24		SBC-Pcons5	1850.25	1009.62	2859.87
25	**6**	**STRUCTFAST-SFST**	1871.85	987.84	2859.69
26		CaspIta	1896.80	947.65	2844.45
27		BAKER-ROBETTA_04	1846.50	974.97	2821.47
28	**7**	**RAPTOR**	1825.75	995.22	2820.97
29	**8**	***BAKER-ROBETTA (Meta)***	1813.15	997.50	2810.65
30		UGA-IBM-PROSPECT	1816.87	990.01	2806.88

Expert hand modeling teams are printed in normal font, while fully automated servers are printed in bold. STRUCTFAST-EXPM was the highest scoring automated server. Given that a total of 124 expert hand modeling teams and 50 automated servers participated in CASP6, by this scoring metric, STRUCTFAST-EXPM outperformed 100% of the automated servers and ≈96% of the hand modeling teams.

CAFASP4 Results

CAFASP is a prediction evaluation experiment that only fully automated servers are allowed to enter.[8] A total of 70 automated prediction servers entered CAFASP4 for evaluation. The various servers in CAFASP4 were ranked according to the "$N - 1$" rule, where N denotes the number of targets in the divisions. The final rank a server achieves in CAFASP is the best rank obtained in any subset of size $N - 1$. In addition to measuring the algorithms' sensitivity on "easy" and "hard" targets, CAFASP4 provides a benchmark on a servers' score specificity. The server's specificity is defined as the number of correct predictions that, according to the reported score, have higher confidence than the first, second, . . ., or 10th false positive. Tables 3–5 summarize the performance of the top ranking autonomous servers in CAFASP4 according to the MaxSubDom measure. The performance of the various servers with respect to other CAFASP scoring measures is very similar and can be accessed at http://www.cs.bgu.ac.il/~dfischer/CAFASP4.

CASP6 Results

The Sixth Critical Assessment of Techniques for Protein Structure Prediction (CASP6) ran in parallel with CAFASP4, and we entered this experiment with the same three algorithms, SFST, BNMX, and EXPM. Because the CASP6 experiment allows expert hand modeling groups to submit models, it enables the direct comparison of fully automated servers with expert hand modeling results. A total of 174 prediction teams participated in CASP6, consisting of 124 expert hand modeling teams and 50 fully automated servers. The evaluation metrics used at CASP6 were different from those used at CAFASP4. CASP traditionally uses the AL0 and GDT measures. In addition, at CASP6, the GDT score assigned to any particular protein model could be penalized if there were "physically impossible" regions in the structure, as judged by visual inspection. Table 6 summarizes STRUCTFAST's rank by the sum of the average GDT scores in the easy and hard homology modeling categories at CASP6. The GDT scores shown were compiled directly from the official CASP6 Web site

TABLE VII. The Ranking of CASP6 Groups Using AL0 Measure

Overall rank	Server rank	CASP6 group name	No. of CASP6 models in the top 20 by AL0
1		KOLINKSI-BUJNICKI	79
2		Jones-UCL	69
3		GeneSilico-Group	60
4	**1**	**STRUCTFAST-SFST**	54
5		BAKER	53
6		Ginalski	51
7		TOME	51
8		Skolnick-Zhang2	50
9		UGA-IBM-PROSPECT	46
10	**2**	**RAPTOR**	44
11		CaspIta	43
12		CBRC-3D	38
13		FISCHER	37
14		CHIMERA	34
15	**3**	**BAKER-ROBETTA (Meta)**	30
16		SAM-T04-hand	29
17		SBC	28
18		Sternberg	27
T19		CAFASP-Consensus	26
T19		MCon	26
T19		BAKER-ROBETTA_04	26
22	**4**	**STRUCTFAST-EXPM**	25
23	**5**	**STRUCTFAST-BNMX**	24
T24	**T6**	**zhousp3**	23
T24	**T6**	**ZHOUSPARKS2**	23
T24	**T6**	**ACE (Meta)**	23
27		CMM-CIT-NIH	21
28		SBC-Pcons5	20
29		SBC-Pmodeller5	19
30		3D-JIGSAW	18

As shown in Table VI, the top 30 homology modeling groups at CASP6 included 22 expert hand modeling groups and 8 automated servers. This table shows the number of CASP6 targets (across all categories) that each of these groups placed in the top 20 according to the number of correctly aligned residues in the model (the AL0 metric). Only 3 out of 124 hand modeling groups in the CASP6 competition produced more high quality alignments than STRUCTFAST-SFST.

(http://predictioncenter.org/casp6/Casp6.html). Table 7 shows a ranking of the top servers according to the number of CASP6 models they placed in the top 20.

As expected, in the CASP6 results, the EXPM and BNMX servers outperformed SFST by the GDT metric due to the reward given to the extra residues modeled from the second highest scoring alignment. Conversely, SFST significantly outperformed the EXPM and BNMX servers according to the AL0 metric, because the extra residues were penalized more often than they were rewarded.

Specific Example 1: CASP6 Target T0197

STRUCTFAST's performance on CASP6 Target T0197 illustrates the importance of implementing a rigorous statistical approach to score normalization in profile–profile methods. STRUCTFAST was the only automated server to predict the correct fold for this target (Fig. 2). Interestingly, STRUCTFAST's highest raw score was assigned to the alignment with 1lgtA (Glyoxalase/Bleomycin resistance protein/Dihydroxybiphenyl dioxygenase fold), which was the consensus prediction among most servers at CASP6. However, the alignment of T0197 to the correct parent structure 1di8C was correctly assigned the lowest Convergent Island Statistics *E*-value ($3e^{-2}$ vs. 9.3, Table 8).

TABLE VIII. STRUCTFAST's Raw Scores and the Corresponding E-Values Computed by the Convergent Island Statistics for Target T0197

Template	Raw score	CIS E-value
1lgtA	66.12	9.3
1mpyA	65.47	2.4
1l9zH	64.70	5.8
1g81A	63.09	9.0
1jh6A	62.44	5.7
1d8iC	59.89	3e-2

1d8iC was the correct template for this target, while 1lgtA was the consensus false positive assigned by numerous automated servers.

CONCLUSION

In summary, STRUCTFAST's blind benchmark study prediction results suggest that its novel, structure-based

dynamic programming and unique approach to computing profile–profile *E*-values enable it to produce homology modeling alignments that are commensurate in quality with those produced by an expert hand modeler. This significant milestone in the field of protein modeling was attained by the top performing servers, including STRUCTFAST, for the first time at CASP6.

REFERENCES

1. Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ. Basic local alignment search tool. J Mol Biol 1990;215:403–410.
2. Pearson WR, Lipman DJ. Improved tools for biological sequence comparison. Proc Natl Acad Sci USA 1988;85:2444–2448.
3. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, Miller W, Lipman DJ. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25: 3389–3393.
4. Karplus K, Karchin R, Barrett C, Tu S, Cline M, Diekhans M, Grate L, Casper J, Hughey R. What is the value added by human intervention in protein structure prediction? Proteins 2001;Suppl 5:86–91.
5. Rychlewski L, Jaroszewski L, Li W, Godzik A. Comparison of sequence profiles. Strategies for structural predictions using sequence information. Protein Sci 2000;9:232–241.
6. Yona G, Levitt M. Within the twilight zone: a sensitive profile–profile comparison tool based on information theory. J Mol Biol 2002;315:1257–1275.
7. Rychlewski L, Fischer D. LiveBench-8: the large-scale, continuous assessment of automated protein structure prediction. Protein Sci 2005;14:240–245.
8. Fischer D, Rychlewski L, Dunbrack RL Jr, Ortiz AR, Elofsson A. CAFASP3: the third critical assessment of fully automated structure prediction methods. Proteins 2003;53:503–516.
9. Vincent JJ, Tai CH, Sathyanarayana BK, Lee B. Assessment of CASP6 predictions for new and nearly new fold targets. Proteins 2005;61:67–83.
10. Wang G, Jin Y, Dunbrack RL Jr. Assessment of fold recognition predictions in CASP6. Proteins 2005;61:46–66.
11. Tress M, Ezkurdia I, Grana O, Lopez G, Valencia A. Assessment of predictions submitted for the CASP6 comparative modeling category. Proteins 2005:61:27–45.
12. Ginalski K, Pas J, Wyrwicz LS, von Grotthuss M, Bujnicki JM, Rychlewski L. ORFeus: detection of distant homology using sequence profiles and predicted secondary structure. Nucleic Acids Res 2003;31:3804–3807.
13. Karlin S, Altschul SF. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. Proc Natl Acad Sci USA 1990;87:2264–2268.
14. Dembo A., Karlin S, Zeitouni O. Critical phenomena for sequence matching with scoring. Ann Prob 1994;22:1993–2021.
15. Sadreyev RI, Grishin NV. COMPASS: a tool for comparison of multiple protein alignments with assessment of statistical significance. J Mol Biol 2003;326:317–336.
16. Poleksic A, Danzer JF, Hambly K, Debe DA. Convergent Island Statistics: a fast method for determining local alignment score significance. Bioinformatics 2005;21:2827–2831.
17. Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J Mol Biol 1993;233:123–138.
18. Robinson AB, Robinson LR. Distribution of glutamine and asparagine residues and their near neighbors in peptides and proteins. Proc Natl Acad Sci USA 1991;88:8880–8884.
19. Shi J, Blundell TL, Mizuguchi K. FUGUE: sequence–structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties. J Mol Biol 2001;310: 243–257.
20. Mizuguchi K, Deane CM, Blundell TL, Overington JP. HOMSTRAD: a database of protein structure alignments for homologous families. Protein Sci 1998;7:2469–2471.
21. Higgins D, Thompson J, Gibson T, Thompson JD, Higgins DG, Gibson TJ CLUSTAL W: improving the sensitivity of progressive-multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic Acids Res 1994;22:4673–4680.
22. Marti-Renom MA, Stuart A, Fiser a, Sánchez R, Melo F, Sali A. Comparative protein structure modeling of genes and genomes. Annu Rev Biophys Biomol Struct 2000;29:291–325.
23. Altschul SF, Bundschuh R, Olsen R, Hwa T. The estimation of statistical parameters for local alignment score distributions. Nucleic Acids Res 2001;29:351–361.
24. Milik M, Kolinski A, Skolnick J. An algorithm for rapid reconstruction of protein backbone from alpha carbon coordinates. Comput Chem 1997;18:80–85.
25. Fischer D. 3D-SHOTGUN: a novel, cooperative, fold-recognition meta-predictor. Proteins 2003;51:434–441.
26. Zhou H, Zhou Y. Single-body residue-level knowledge-based energy score combined with sequence-profile and secondary structure information for fold recognition. Proteins 2004;55:1005–1013.
27. Zhou H, Zhou Y. Fold recognition by combining sequence profiles derived from evolution and from depth-dependent structural alignment of fragments. Proteins 2005:58;321–328.
28. Ginalski K, von Grotthuss M, Grishin NV, Rychlewski L. Detecting distant homology with Meta-BASIC. Nucleic Acids Res 2004; 32:W576–W581.