HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment

Michael Remmert, Andreas Biegert, Andreas Hauser & Johannes Söding

Sequence-based protein function and structure prediction depends crucially on sequence-search sensitivity and accuracy of the resulting sequence alignments. We present an open-source, general-purpose tool that represents both query and database sequences by profile hidden Markov models (HMMs): 'HMM-HMM–based lightning-fast iterative sequence search' (HHblits; http://toolkit.genzentrum.lmu.de/hhblits/). Compared to the sequence-search tool PSI-BLAST, HHblits is faster owing to its discretized-profile prefilter, has 50–100% higher sensitivity and generates more accurate alignments.

Building protein multiple-sequence alignments (MSAs) by iterative sequence searches is of fundamental importance in computational biology, as MSAs are a key intermediate step in the sequence-based prediction of evolutionarily conserved properties, such as tertiary structure, functional sites or interaction interfaces. Sequence profiles and profile hidden Markov models (HMMs) are condensed representations of MSAs that specify for each sequence position the probability of observing each of the 20 amino acids in evolutionarily related proteins. PSI-BLAST[1], the most widely used iterative search tool, progressively refines a query sequence profile by adding statistically significant sequence matches to the profile for the next search iteration. The tools SAM2K (ref. 2) and HMMER3 (ref. 3) use profile HMMs for better sensitivity.

Profile-profile and HMM-HMM alignment are the most sensitive classes of sequence-search methods. They are the methods of choice for identifying and aligning templates for three-dimensional homology modeling[4]. Our HMM-HMM alignment method HHsearch[5] is used by many of the best protein structure prediction servers, among which is HHpred[6], the top-ranked server for template-based protein structure prediction in last year's Critical Assessment of Techniques for Protein Structure Prediction exercise (http://predictioncenter.org/casp9/groups_analysis.cgi?type=server&tbm=on/). However, these methods are generally too slow for iteratively searching through large sequence databases such as UniProt or NCBI's nonredundant (nr) database. Here we present HMM-HMM–based lightning-fast iterative sequence search (HHblits), which extends HHsearch to enable fast, iterative sequence searches. The profile-profile alignment prefilter of HHblits reduces the number of full HMM-HMM alignments from many millions to a few thousand, making it faster than PSI-BLAST but still as sensitive as HHsearch (**Supplementary Fig. 1**).

For iterative searches, HHblits needs a database of HMMs that covers the entire sequence space. We devised a very fast method, kClust (M. Hauser, C.E. Mayer and J.S., unpublished data), for clustering large sequence databases down to 20–30% maximum pairwise sequence identity while requiring almost full-length alignability (>80% coverage of longer sequences). This strict coverage criterion enriches for orthologous sequences with the same domain architecture[7]: of the UniProt20 clusters containing more than two Swiss-Prot sequences with enzyme commission numbers, 98.4% had all four enzyme commission digits conserved (**Supplementary Fig. 2**). kClust is sufficiently fast (~1,000 times faster than BLAST) to allow for regular reclustering of the updated UniProt and nr databases. UniProt20 (the version from July 2011) contained 15 million sequences in 2.6 million HMMs, with an average of 5.5 sequences per cluster.

HHblits first converts the query sequence (or MSA) to an HMM. This is conventionally done by adding pseudocounts of amino acids that are physicochemically similar to the amino acid in the query. In contrast, HHblits calculates pseudocounts that depend on the local sequence context (that is, the 13 positions around each residue). This method had improved the sensitivity and alignment quality of the resulting profile considerably[8]. HHblits then searches the HMM database and adds the sequences from HMMs below a defined expected value (*E* value) threshold to the query MSA, from which the HMM for the next search iteration is built (**Fig. 1a** and **Supplementary Fig. 3**). For speed and sensitivity, the prefilter is crucial. The key idea was to implement profile-profile comparison as a sequence-to-profile comparison by discretizing the vectors of 20 amino acid probabilities in each HMM column into an alphabet of 219 letters. Each letter represents a typical profile column (**Supplementary Fig. 4**). We approximate the database HMMs by sequences over this extended alphabet, ignoring the insertion and deletion probabilities of the HMMs (**Supplementary Fig. 5**). Before prefiltering, we calculate the score of each query HMM column with each of the 219 letters, which results in a 219-row extended sequence profile. The prefiltering consists of two steps (**Supplementary Fig. 3**): (i) a very fast gapless local alignment between the extended query profile and the extended database sequences and (ii) a gapped

Gene Center and Center for Integrated Protein Science Munich, Ludwig-Maximilians Universität München, Munich, Germany. Correspondence should be addressed to J.S. (soeding@genzentrum.lmu.de).

RECEIVED 29 JULY; ACCEPTED 1 DECEMBER; PUBLISHED ONLINE 25 DECEMBER 2011; DOI:10.1038/NMETH.1818

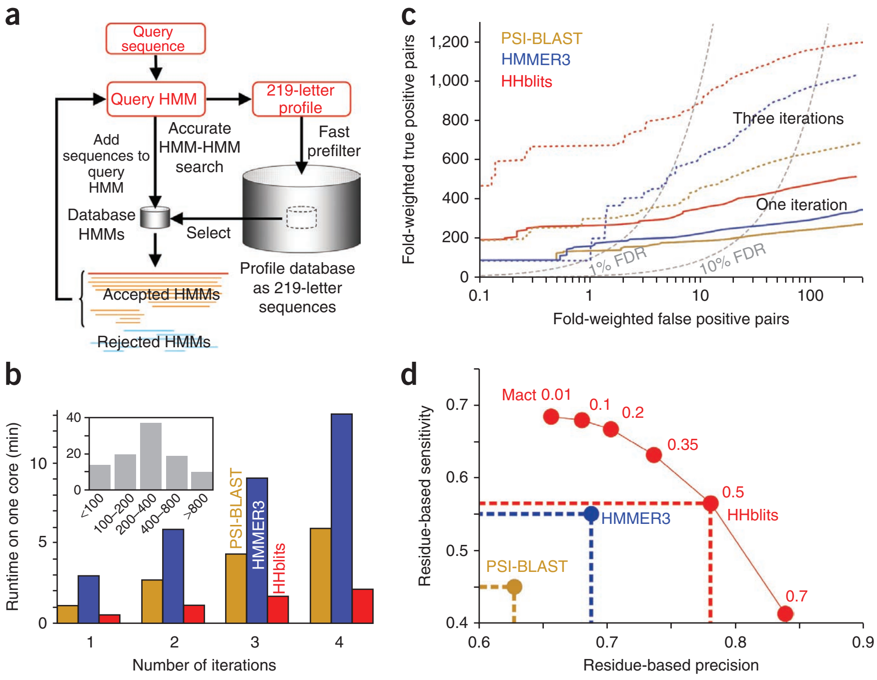

Figure 1 | Workflow and benchmark comparison. (**a**) HHblits can iteratively search for homologous sequences in large databases such as UniProt. The HHblits database is a clustered version in which each set of full-length alignable sequences is represented by an HMM. Sequences from matched HMMs with a statistically significant *E* value are added to the query MSA, from which a new HMM is calculated for the next search iteration. A prefilter reduces the number of full HMM-HMM alignments by ~2,500-fold. (**b**) Median run times for searches with 100 test sequences through the UniProt or UniProt20 database (the inset shows the test sequence length distribution). (**c**) True positive pairs (same SCOP fold) compared to false positive pairs (different SCOP fold) for one and three search iterations in an all-against-all comparison. FDR, false discovery rate. (**d**) Mean fraction of correctly aligned residue pairs out of all structurally alignable pairs (sensitivity) compared to the fraction of correctly aligned pairs out of all the aligned pairs (precision). The parameter mact controls the alignment greediness (**Supplementary Fig. 10**).

local alignment. For step ii, we modified the code from previous work[9]. Each of the two steps allows 1–5% of the sequences to pass. We implemented both filters with streaming SIMD extension 3 (SSE3) instructions, which are available on all modern Intel and Advanced Micro Devices (AMD) central processing units and process 16 single-byte operations per core and clock cycle[9]. The database HMMs whose extended sequences passed the prefilter are aligned to the query HMM, and *E* values are calculated. Statistically significant matches are realigned with a local maximum accuracy algorithm[10].

A single search iteration with HHblits version 2.2.17 through UniProt20 (2.6 million clusters and 15 million sequences) for 100 randomly selected query sequences took a median 31 s and an average 1 min 13 s on a single Xeon 2.9 GHz core (**Fig. 1b** and **Supplementary Data 1**). For a single search iteration through UniProt (15 million sequences), PSI-BLAST needed 1 min 7 s (median) and 1 min 26 s (average) and HMMER3 needed 2 min 57 s (median) and 5 min 8 s (average). Additional iterations took roughly the same amount of time as the first iteration (**Supplementary Fig. 6**), and therefore overall, HHblits was about twice (15%) as fast as PSI-BLAST and was 6× (median) and 4× (average) faster than HMMER3.

We compared the sensitivity of HHblits to that of PSI-BLAST and HMMER3 in detecting homologous proteins (to rank true positive, homologous pairs above false positive, unrelated pairs) (**Fig. 1c**). We performed an all-against-all comparison of 5,287 representative domain sequences from the Structural Classification of Proteins (SCOP) database[11]. After one iteration, HHblits detected 107% more true positive pairs than PSI-BLAST and 53% more than HMMER3 at 1% false discovery rate, and after three iterations, the improvement was 147% over PSI-BLAST and 69% over HMMER3. We obtained similar values in a receiver operating curve 5 (ROC5) analysis (Online Methods and **Supplementary Fig. 7**). Furthermore, HHblits reported more reliable *E* values than PSI-BLAST (**Supplementary Fig. 8**).

To assess the quality of the pairwise alignments (**Fig. 1d**), we randomly selected from each SCOP superfamily up to ten pairs of domains with <30% sequence identity and a TM-align (Online Methods) structural similarity score of >0.6 (**Supplementary Data 2**). For each method, we built MSAs for the queries using

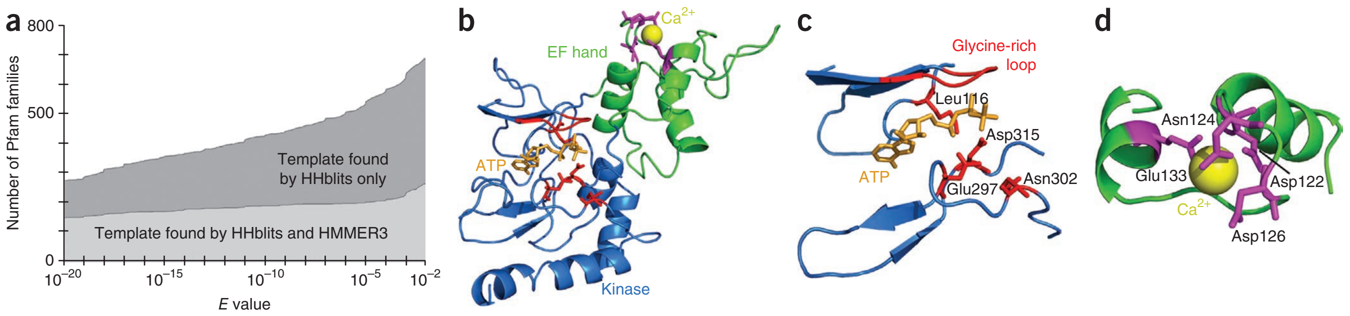

Figure 2 | Structure predictions for Pfam families and the modeling of human Pip49 (also known as FAM69B). (**a**) Families to which only HHblits and both HHblits and HMMER3 assigned a structural template below a given *E* value. (**b**) Homology model of human Pip49 kinase domain (blue) with the inserted EF hand (green). (**c**) Catalytic center showing the conserved residues (red) for protein kinase activity. (**d**) EF hand insertion with the conserved residues (magenta) for the predicted Ca^{2+}-dependent activation.

two search iterations through UniProt and aligned the resulting query MSAs with their corresponding templates. We determined correctly aligned residues through comparison with the structural alignments. Compared to PSI-BLAST and HMMER3, HHblits sensitivity per residue using default parameters (mact 0.5) was 12 and 2 percentage points higher and the precision per residue was 15 and 10 percentage points higher, respectively (**Fig. 1d**). The higher precision of HHblits alignments explains its robustness against homologous overextension (tested on a benchmark with multidomain proteins; **Supplementary Fig. 9**), which is the main cause of corrupted PSI-BLAST alignments[12].

As another measure of MSA quality, we sought to improve the accuracy of PSIPRED[13] secondary structure prediction by running PSIPRED on MSAs generated by HHblits. Although PSIPRED had been trained on PSI-BLAST MSAs, HHblits MSAs improved the Q3 score (fraction of correctly predicted secondary structure states) for proteins from the PDBselect 2007 dataset (Online Methods) from 80.4% to 81.3% and the secondary structure segment overlap (SOV) score from 77.5% to 78.6% (**Supplementary Table 1**). These results, obtained without training a large parameter set, are among the best achieved at present[14].

A potential drawback of HHblits is the requirement that its databases consist of MSAs and their HMMs instead of single sequences. Although we will regularly update standard HHblits databases such as UniProt20, nr20, PDB, SCOP and Pfam, customized databases, for example databases representing an organism's proteome, will need to be built specifically for HHblits.

To show the utility of HHblits, we predicted structures for Pfam families[15] for which no template is known and also for which no template is known for any family from its Pfam clan (**Fig. 2a**). We jumpstarted two HHblits iterations through UniProt20 with the Pfam seed alignment and then searched the PDB70 database (ftp://toolkit.genzentrum.lmu.de/HHblits/databases/). HHblits assigned templates to 620 families with $E < 10^{-3}$, only 226 of which HMMER3 detected (41 families were found only by HMMER3 and not HHblits) (**Supplementary Table 2**).

As an example of these results, we describe the predictions for Pip49-C, the C-terminal part of the pancreatitis induced protein 49, a Pfam domain of unknown structure and function with a predicted N-terminal transmembrane helix. The 100 best HHblits matches in PDB70 were with protein kinases (best *E* value of 2×10^{-20}), even though the Pfam MSA is missing 70 N-terminal residues from the kinase domain. An HHblits search started with full-length human Pip49 (also known as FAM69B) (with two iterations through UniProt20 and one iteration through PDB70) detected many protein kinase domains, and, notably, a tandem Ca^{2+}-binding EF hand (*E* value = 0.09) inserted in the kinase domain. Based on our homology models (**Fig. 2b–d** and **Supplementary Data 3**) and the conservation of key residues, we predict that Pip49 and its paralog FAM69A are membrane-bound protein kinases in the lumen of the endoplasmic reticulum that are activated by Ca^{2+} through structural rearrangement of their EF hand.

In conclusion, HHblits is an open-source, robust, general-purpose, iterative protein sequence search tool that is faster, considerably more sensitive and produces alignments of much better quality than PSI-BLAST. HHblits has the potential to improve many downstream analysis and prediction methods, such as a *de novo* protein structure prediction method requiring large and accurate MSAs[16].

METHODS

Methods and any associated references are available in the online version of the paper at http://www.nature.com/naturemethods/.

Note: Supplementary information is available on the Nature Methods website.

ACKNOWLEDGMENTS

We acknowledge financial support by the Deutsche Forschungsgemeinschaft (grant SFB646) and by a Gastprofessur grant from Ludwig-Maximilians Universität Munich financed through the Excellence Initiative of the Bundesministerium für Bildung und Forschung.

AUTHOR CONTRIBUTIONS

M.R. performed research, J.S. initiated and guided research, A.B. generated the profile-column alphabet, A.H. contributed code for fast file access, and M.R. and J.S. wrote the manuscript.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

Published online at http://www.nature.com/naturemethods/.
Reprints and permissions information is available online at http://www.nature.com/reprints/index.html.

1. Altschul, S.F. *et al. Nucleic Acids Res.* **25**, 3389–3402 (1997).
2. Karplus, K., Barrett, C. & Hughey, R. *Bioinformatics* **14**, 846–856 (1998).
3. Eddy, S.R. *Genome Inform.* **23**, 205–211 (2009).
4. Söding, J. & Remmert, M. *Curr. Opin. Struct. Biol.* **21**, 404–411 (2011).
5. Söding, J. *Bioinformatics* **21**, 951–960 (2005).
6. Söding, J., Biegert, A. & Lupas, A.N. *Nucleic Acids Res.* **33**, W244–W248 (2005).
7. Hegyi, H. & Gerstein, M. *Genome Res.* **11**, 1632–1640 (2001).
8. Biegert, A. & Söding, J. *Proc. Natl. Acad. Sci. USA* **106**, 3770–3775 (2009).
9. Farrar, M. *Bioinformatics* **23**, 156–161 (2007).
10. Biegert, A. & Söding, J. *Bioinformatics* **24**, 807–814 (2008).
11. Andreeva, A. *et al. Nucleic Acids Res.* **36**, D419–D425 (2008).
12. Gonzalez, M.W. & Pearson, W.R. *Nucleic Acids Res.* **38**, 2177–2189 (2010).
13. Jones, D.T. *J. Mol. Biol.* **292**, 195–202 (1999).
14. Aydin, Z., Singh, A., Bilmes, J. & Noble, W. *BMC Bioinformatics* **12**, 154 (2011).
15. Finn, R.D. *et al. Nucleic Acids Res.* **38**, D211–D222 (2010).
16. Marks, D.S. *et al. PLoS ONE* **6**, e28766 (2011).

ONLINE METHODS

HHblits server usage in a nutshell. The HHblits server (http://hhblits.genzentrum.lmu.de/ or http://toolkit.genzentrum.lmu.de/hhblits/) takes as input a single sequence or an MSA and iteratively searches through the selected HMM databases (UniProt20, nr20, PDB, SCOP and Pfam) for a specified number of iterations. Two or more iterations only make sense when a database covering the entire sequence space (such as UniProt20 or the nr20) is selected. A larger number of search iterations increases the sensitivity of the alignment but also increases the risk of alignment corruption, for example, through homologous overextension[12]. Owing to this trade-off, we recommend between one and four iterations. For optimum reliability it is advisable to first identify domains in the query sequence by performing a single HHblits iteration through the PDB70 database and then to cut the query sequence along domain boundaries into shorter segments, which are less prone to alignment corruption.

When the "realign with MAC" box is checked in HHblits, it calculates the more accurate maximum accuracy (MAC) HMM-HMM alignments after the Viterbi HMM-HMM comparisons. However, the Viterbi alignments are better suited for the calculation of scores, *E* values and probabilities, and, therefore, the results are a combination of Viterbi scores and *E* values and MAC alignments. The MAC threshold parameter 'mact' controls the alignment greediness during MAC realignment. At a mact value of 0.35, segments that have an average probability of being correct of below 35% will be omitted. A mact value of 0.01 will generate quasi-global MAC alignments for use in, for example, homology modeling, while still using the local Viterbi alignment for the scoring. When searching with single domains, a mact value of 0.2 can be sufficient; otherwise, higher mact values (for example, 0.5) are recommended. The MSA built by HHblits can be inspected and extracted under the "show alignment" tab on the results page. An MSA of consensus sequences of matched HMMs can be viewed with a Jalview applet on the results page, for example to check for alignment corruption. For more information, see the HHblits server help pages and the HHblits user guide.

HHblits command line usage in a nutshell. HHblits is available as source code and as executable RPM and DPKG packages for most Linux 64 bit platforms, MAC OS X and Berkeley Software Distribution (BSD) Unix at http://hhblits.genzentrum.lmu.de/. The command "$ hhblits -i query.fasta -d /databases/UniProt20 -n 2 -mact 0.01 -oa3m query_msa.a3m" will run two search iterations through the UniProt20 database, starting from the input sequence (or MSA) in query.fasta. The mact value 0.01 generates quasi-global alignments. The human-readable output is written to query.hhr by default (this option can be changed using -o <file>), whereas the resulting MSA is written to query_msa.a3m in a3m format. This format can be transformed to other formats using reformat.pl. Custom databases (such as for a single genome) can be built by generating MSAs for each protein sequence using, for example, two HHblits iterations, adding secondary structure with the Perl script addss.pl and building the HHblits database files using create_db.pl and create cs_db.pl. For more information see the user guide in the HHblits package at http://hhblits.genzentrum.lmu.de/ or http://toolkit.genzentrum.lmu.de/hhblits/.

Fast sequence clustering with kClust. We developed kClust to cluster large sequence databases for use with HHblits in a fraction of the time that would be necessary using BLAST and down to much lower sequence identities (20–30%) than is possible with CD-HIT[17]. kClust achieves its high speed and sensitivity with two new algorithms. First, a fast prefilter sums the similarity scores of all similar 6-mers between sequences *Q* and *T*. The score threshold is set stringently such that only ~$4 \times 10^{-6} \times L_Q L_T$ chance matches occur between sequences of lengths L_Q and L_T. Thus, the time to compare two sequences is reduced in comparison to classic dynamic programming approaches by a factor of ~2.5×10^5. A more sensitive comparison is performed in the second step using dynamic programming on the set of similar 4-mers between *Q* and *T*. Here the threshold is set such that chance matches occur with a probability of ~2×10^{-3} per 4-mer pair. This allows us to achieve a speedup relative to the SSEARCH implementation of classic Smith-Waterman dynamic programming by a factor ~30. kClust binaries and scripts for automatically generating MSAs from clusters are available at ftp://toolkit.genzentrum.lmu.de/kClust/.

Discretized profile-column alphabet. We discretized profile columns into an alphabet of 219 states (the number of printable ASCII characters), where each letter represents a typical profile column. This allows us to approximate any sequence profile by a sequence over this 219-letter extended alphabet. To compare two profiles, we first calculate the score S_{ik} of each query profile column *i* with each of the 219 letters *k*

$$S_{ik} = \log_2 \sum_{a=1}^{20} q_i(a) p_k(a) / f(a)$$

where $q_i(a)$ denotes the query profile at position *i*, $p_k(a)$ is the profile column represented by the letter $k \in \{1,\ldots,219\}$ and $f(a)$ is the background frequency of residue *a*. We thus obtain a 219-row extended sequence profile, which can be aligned to extended sequences representing the other profile using fast, standard dynamic programming techniques. We generated the 219-letter alphabet using the same method that was previously used for learning an optimal set of sequence context profiles[8], but here we set the window size from 13 to 1 residue. We also set the window weights w_j to 100 to obtain a hard clustering. We initialized the 219 states randomly and maximized the likelihood that the 10 million training sequence profile columns were generated by the 219 profile columns. The best of several trials was used. The 10 million profile columns were randomly sampled from the MSAs in our clustered nonredundant database.

Pre-filtering. In the two prefilter steps, the extended query sequence profile is aligned to the extended database sequences. The first step calculates the score of the largest ungapped alignment. To pass this filter, the score has to be larger than 2.5 + $\log_2(L_Q L_T)$ bits, where L_Q and L_T are the lengths of the query profile and database sequence, respectively. The log term is a standard length correction. The second step calculates a Smith-Waterman alignment with affine gap penalties (gap open: 5 bits, gap extend: 1 bit). From the bit score *S*, an approximate *E* value is calculated: $E = N_{\mathrm{db}}\, L_Q L_T \times 2^{-S}$, where N_{db} is the number of sequences per HMMs in the database, and sequences pass if their *E* value is

doi:10.1038/nmeth.1818

below the prefilter threshold (E_{pre}) = 1,000. Each filter step leads to a 10- to 100-fold reduction of database sequences.

Both filters were implemented with SSE3 instructions that process 16 single bytes in parallel on 128-bit SIMD units present on each central processing unit (CPU) core. Each byte holds the score in units of 1/4 bits plus an offset of 50, which allows us to represent a score range between −12.5 and +51.5 bits. The algorithms were programmed such that the scores will saturate at 255 on overflow. Because any score larger than 51 bits will always pass the filter, this range is sufficient for prefiltering. The first step processes four or five cells of the dynamic programming matrix per CPU clock cycle, and the second step processes ~1.3 cells per clock cycle. The clustered UniProt database (version from 07/2011) contains 2.6 million sequences of average length 320 cells, and therefore the first prefilter search with a query profile of length 300 through UniProt takes about $300 \times 320 \times 2.6 \times 10^6$ / (4.5×2.9 GHz) = 18 s, which is about 25% of the average time needed for the entire HHblits search.

For sequences that pass the two prefilters, we calculate local alignments using SSE3 instructions to restrict the resulting HMM-HMM alignment to the region likely to contain the true alignments. For back-tracing, we need to prevent the score from saturating. Therefore, each score is held in 2 bytes in this step (again, in units of 1/4 bits), which yields a score range of −12.5 bits to +16,371.5 bits. Up to ten suboptimal alignments are extracted by masking all cells at a distance of <150 residues from the previously extracted alignments until the prefilter E value is above the E_{pre} value.

Viterbi alignment and *E* value calculation. To speed up the time-consuming HMM-HMM alignment steps, all cells with a distance of >200 residues to all alignments identified in the previous step are masked out. An HMM-HMM alignment is performed on the active cells using the Viterbi algorithm from HHsearch. The Viterbi algorithm determines the alignment with the maximum score. Even though it does not yield the most accurate alignments (see the maximum accuracy alignment section below), it yields reliable scores for ranking and P value calculation. From the Viterbi score S, a P value is calculated using an extreme value distribution: $P = 1 - \exp(-\exp(-\lambda\ (S - \mu)))$. The extreme value distribution parameters μ and λ are estimated from the four features L_Q, L_T, N_Q^{eff} and N_T^{eff} using two standard two-layer neural networks with four hidden nodes each. Here N_Q^{eff} and N_T^{eff} are the numbers of effective sequences in the query and template HMMs, respectively (defined in ref. 5). The Viterbi E value is calculated from the P value using $E = N_{\mathrm{db}}\ P\ (E_{\mathrm{pre}}/N_{\mathrm{db}})^{\alpha}$, where $\alpha = 0.4 + 0.02 \times (N_T^{\mathrm{eff}} - 1) \times (1 - 0.1 \times (N_Q^{\mathrm{ff}} - 1))$. The term $(E_{\mathrm{pre}}/N_{\mathrm{db}})^{\alpha}$ is an empirical correction for the correlation between the prefiltering and Viterbi scores ($\alpha = 0$: perfect correlation, $\alpha = 1$: no correlation). The three coefficients for α were optimized to yield accurate E values (**Supplementary Fig. 8**).

Further speedups. Viterbi alignments are performed in the order of decreasing prefilter E values. We stop the time-consuming HMM-HMM comparisons when very few homologs are likely to have been observed among the last 200 HMM-HMM alignments. A coarse estimate for the probability of a match to be a true homolog is $1/(1 + E)$ for a Viterbi E value of E. We average $1/(1 + E)$ over the last 200 processed Viterbi alignments and skip all further database HMMs when this average drops below 0.01.

Maximum accuracy alignment. Whereas the Viterbi algorithm calculates the alignment with the best score, the maximum accuracy alignment (proposed in ref. 18) yields the global alignment with the maximum possible accuracy as defined by the sum of probabilities for each residue pair to be correctly aligned

$$\sum\nolimits_{(i,j)\in \mathrm{alignment}} P(i\,\text{aligned to }j) \rightarrow \max$$

We extended this algorithm to local HMM-HMM comparison[10], which produces the local alignment that maximizes the sum of probabilities for each residue pair to be correctly aligned minus the mact penalty

$$\sum\nolimits_{(i,j)\in \mathrm{alignment}} \left(P(i\,\text{aligned to }j) - \mathrm{mact}\right) \rightarrow \max$$

With the mact parameter, the alignment greediness can be controlled from nearly global, long, greedy alignments (mact near 0) to very precise and short alignments (mact near 1). To speed up the MAC alignment, cells at a city block distance of >200 from the optimal and all suboptimal Viterbi HMM-HMM alignments are masked.

Adding sequences from significant matches to the query HMM. Sequences from all HMMs below the Viterbi E value inclusion threshold (with a default value of 10^{-3}) are read from the alignment files of the clustered database and aligned to the query MSA according to the HMM-HMM maximum accuracy alignment. The query HMM is calculated from the query MSA.

Parameter optimization. We optimized the parameters (filter thresholds, gap costs, amino acid and transition pseudocount strengths and E value inclusion threshold) on an optimization set that had no member from the same fold as the sequences in the test set (see the sensitivity benchmarks section below). We varied the parameters in discrete steps one after another, performed an all-against-all search on the optimization set and tried to maximize the mean ROC5 value (see below). For the prefilter settings, we chose the best trade off between efficiency and sensitivity.

Sensitivity benchmarks. We filtered the sequences from SCOP 1.73 (ref. 11) to a maximum pairwise sequence identity of 20%. We assigned every fifth fold to the optimization set (1,329 sequences in 215 folds) and the other folds to the test set (5,287 sequences in 862 folds; **Supplementary Data 4**). SCOP is a hierarchically ordered database of protein domain sequences with known structure. We considered domains from the same fold as true positive, homologous pairs and domains from different folds as false positive, nonhomologous pairs. Exceptions to this were members of Rossman-like folds (c.2–c.5, c.27, c.28, c.30 and c.31) and the four- to eight-bladed β-propellers (b.66–b.70), which are probably related and which we treated as 'unknown'. To prevent a few large folds from dominating the benchmark[4], we weighed each hit with the value of one over the number of members in the query SCOP fold ('fold-weighted true positives and false positives'). All but the last search iteration were performed against the UniProt database. The final iteration of PSI-BLAST and HMMER3 searches were performed against all UniProt and SCOP sequences. For HHblits, the final iteration was performed against the UniProt20/SCOP database, a UniProt20 database to which SCOP test sequences had been added as singleton clusters: each SCOP sequence from the test set was either mapped to its UniProt20 cluster containing the test sequence or was added as a singleton cluster to UniProt20/SCOP if no matching cluster was found. All pairs of domains were ranked by *E* value for each of the tools, and the number of true positives versus false positives below a given *E* value were plotted. The ROC5 plots in **Supplementary Figures 7d** and **9b** assess how well a method ranks the matched proteins within each search. These plots show the fraction of queries with ROC5 scores above the threshold on the *x* axis. The ROC5 score is the area under the true positive versus false positive ROC curve up to the fifth false positive divided by the area under the optimal ROC curve.

Sensitivity benchmark for multidomain proteins. Because multi-domain protein sequences present particular challenges, such as homologous overextension[12], to iterative sequence search methods, we tested our tools on a benchmark set of multi-domain proteins. For each of the 5,287 sequences in our test set, we searched for a sequence in the nonredundant database that had a BLAST match to the SCOP sequence with an *E* value $<10^{-40}$, sequence coverage >95% sequence identity >60% and whose full-length sequence contained at least 100 additional residues. These criteria resulted in 2,343 multidomain proteins. For all extracted multi-domain proteins, we proceeded as described in the previous paragraph (two iterations through UniProt and one iteration through UniProt or UniProt/SCOP, respectively). We counted true positive and false positive pairs only if the alignment covered at least 50 residues of the SCOP domain in the nonredundant query sequence.

Alignment quality. To assess the accuracy of the pairwise alignments (**Fig. 1d**), we chose 4,128 query-template pairs by randomly selecting from each SCOP superfamily up to ten pairs with <30% sequence identity and a structural similarity TM-align[19] score >0.6 (**Supplementary Data 2**). For each method, we built MSAs for the queries using two search iterations through UniProt and aligned the resulting query MSAs with their corresponding templates. For HHblits, we selected the template HMMs from the clustered UniProt20 that contained the SCOP template sequence (using the same procedure as described in section Sensitivity benchmarks). We determined correctly aligned residues by comparison with structural alignments from TM align[19].

Improving PSIPRED secondary structure prediction. We compared the accuracy of the secondary structure prediction of PSIPRED[13] using the PSIPRED procedure to generate sequence profiles (three iterations of PSI-BLAST on a filtered database), with the accuracy of PSIPRED run on profiles built from MSAs generated by HHblits. As a test set, we used PDBselect 2007 (ref. 20), which contains 3,649 sequences ranging from 30 to 1,040 amino acids in length. We built MSAs for each sequence using two and three iterations of PSI-BLAST on the nr database filtered with pfilt from the PSIPRED package and using one, two and three iterations of HHblits through UniProt20. HHblits alignments with a diversity around 7.0 were generated by applying hhfilter from the HHblits package with the option '-neff 7'. For all MSAs, we performed the PSIPRED procedure with the default parameters and calculated the Q3 and SOV scores based on the known DSSP sequences (mapping E and B to strand, H, G and I to helix, and S, T and C to coil states).

Fold prediction for Pfam. For nearly half of all Pfam families in version 24.0 (5,716 out of 11,913), no structure is known, and the structures of any of the remotely related families in their Pfam clan are also unknown. We generated MSAs for these 5,716 Pfam families by using their seed alignments as input and performing two iterations with HHblits through the UniProt20 database. The PDB70 database of HHpred was searched with the resulting MSAs. For HMMER3, we scanned the PDB70 sequence database with the full HMMER3 models provided by Pfam.

Pip49/FAM69B modeling. We built an MSA for human Pip49/FAM69B (UniProt identifier Q5VUD6) by running two iterations of HHblits through the clustered UniProt database and adding the secondary structure prediction from PSIPRED to this MSA using the script addss.pl from the HHblits package (**Supplementary Data 5**). To identify structural homologs, the PDB database was scanned by HHblits with this MSA with a mact value of 0.2. From the list of PDB matches, we chose as templates a protein kinase with bound ATP (PDB identifier 1RDQ) and a Ca^{2+}-bound EF hand (PDB identifier 3C1V). We used the corresponding HHblits alignments to create a homology model with MODELLER[21] (**Supplementary Data 3**). Although many protein kinases contain EF hands downstream of their kinase domains[15], Pip49 is the first one known in which an EF hand is inserted in the kinase domain, directly after the small N-terminal β sheet. We validated the presence of the EF hand insertion by building an MSA with two iterations of HHblits starting from the presumed inserted sequence and then searching the PDB70 database. This yielded highly significant matches with EF hands (best *E* value = 4×10^{-5}). The previously reported transmembrane helix from residue position 31 to 51 could be confirmed by HMMTOP, MEMSAT-SVM and Phobius. The kinase domain is framed by two short domains with highly conserved cysteines that are likely to form disulfide bonds, which suggests that it resides in the lumen of the endoplasmic reticulum.

17. Li, W. & Godzik, A. *Bioinformatics* **22**, 1658–1659 (2006).
18. Holmes, I. & Durbin, R. *J. Comput. Biol.* **5**, 493–504 (1998).
19. Zhang, Y. & Skolnick, J. *Nucleic Acids Res.* **33**, 2302–2309 (2005).
20. Griep, S. & Hobohm, U. *Nucleic Acids Res.* **38**, D318–D319 (2009).
21. Sali, A. & Blundell, T.L. *J. Mol. Biol.* **234**, 779–815 (1993).

doi:10.1038/nmeth.1818