JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

pubs.acs.org/jcim Article

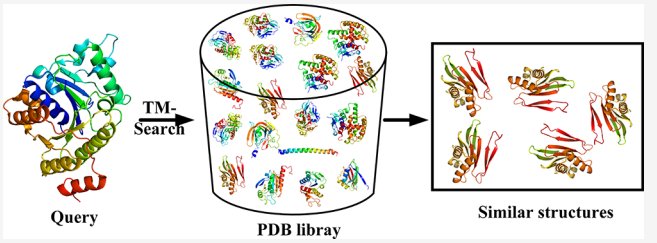
TM-search: An Efficient and Effective Tool for Protein Structure Database Search

Zi Liu,[#] Chengxin Zhang,[#] Qidi Zhang, Yang Zhang,[*] and Dong-Jun Yu[*]

Cite This: *J. Chem. Inf. Model.* 2024, 64, 1043−1049 Read Online

ACCESS | Metrics & More | Article Recommendations

ABSTRACT: The quickly increasing size of the Protein Data Bank is challenging biologists to develop a more scalable protein structure alignment tool for fast structure database search. Although many protein structure search algorithms and programs have been designed and implemented for this purpose, most require a large amount of computational time. We propose a novel protein structure search approach, TM-search, which is based on the pairwise structure alignment program TM-align and a new iterative clustering algorithm. Benchmark tests demonstrate that TM-search is 27 times faster than a TM-align full database search while still being able to identify ∼90% of all high TM-score hits, which is 2−10 times more than other existing programs such as Foldseek, Dali, and PSI-BLAST.



INTRODUCTION

Protein structure alignment is fundamental to many applications in computational structure biology, including structure-based function annotation,[1,2] structure refinement,[3] and protein design.[4] Many of these applications require a search of a query structure through databases of structures. Due to the ever-increasing size of the Protein Data Bank (PDB)[5] and, more recently, databases of computationally predicted structures such as the AlphaFold database,[6] the alignment of a query structure through the whole structure database has become computationally prohibitory. For example, as of the writing of this manuscript, the PDB database houses ∼400,000 protein chains from 178,000 structures (June 2020). Such a large database would take TM-align,[7] which is one of the fastest structure alignment programs, and several CPU days to search a query PDB as TM-align on average generates a pairwise alignment in 0.5 s. Therefore, it is urgent to develop a fast and accurate algorithm for searching the whole protein structure database within reasonable time.

There are two main approaches to accelerate structure database searches. The first is to improve the speed of each pairwise structure alignment by reducing the 3D structure of a protein into a one-dimensional descriptor of structure features, as implemented by Foldseek,[8,9] DeepFold,[10] ContactLib,[11] 3D-AF-SURFER,[12] GraSR,[13] MADOKA,[14] and GR-Align.[15] While such a reformulation of the 3D structure to 1D structure descriptors, also known as structural fingerprints, enables very fast database searches, the conversion also results in the loss of a significant amount of structure information. This is probably the reason why algorithms relying on structure fingerprints have limited sensitivity for remote homologue detection compared to full-fledge 3D structure alignment programs, as shown in the Results and Discussion section.

Another approach for a faster structure database search is to reduce the inherent redundancy of the database by clustering database entries. Pairwise structure alignment can then be performed only between the query structure and the representative structures from each cluster. This can be optionally followed by an alignment to nonrepresentative members if the representative is found to have a high similarity to the query structure. Traditionally, database clustering is performed based on sequence similarity: for example, both the Dali[16] and FATCAT[17] servers offer database search of PDB clustered at a 90% sequence identity cutoff. One issue with the sequence clustering strategy is that many proteins with similar structures but different sequences are grouped into different clusters (i.e., MMseqs2[18] and CD-HIT[19]) as structures are often far more evolutionary conserved than sequences. This phenomenon limits the extent to which sequence clustering reduces the database size. Recently, it was proposed that a structure database can be clustered by structure similarity, such as that implemented by mTM-align[20] and PhyreStorm.[21] This offers a greater compression compared to sequence clustering while still retaining all representative structures for structure search purposes.

Received: September 10, 2023
Revised: January 16, 2024
Accepted: January 16, 2024
Published: January 25, 2024

https://doi.org/10.1021/acs.jcim.3c01455
J. Chem. Inf. Model. 2024, 64, 1043−1049

Inspired by these pioneering works, we developed TM-search, a tool for an efficient and reliable structure search against the entire PDB. TM-search aims to identify every similar structure from the PDB above a user-defined TM-score[22] for a given query structure. TM-search uses a hierarchical database for protein similarity structure search and avoids using the time-consuming all-against-all mode of the protein structure database. The hierarchical database was clustered by the TM-score similarity matrix. When using a query structure to search the whole protein structure database, the representatives of each cluster are aligned to the query, and all the clusters with similarities below the TM-score threshold will be discarded. This is followed by the alignment between the members of each remaining cluster and the query structure. Therefore, this process can theoretically obtain all of the similar structures of the query and greatly improve database search speed. The flowchart of the TM-search is illustrated in Figure 1.

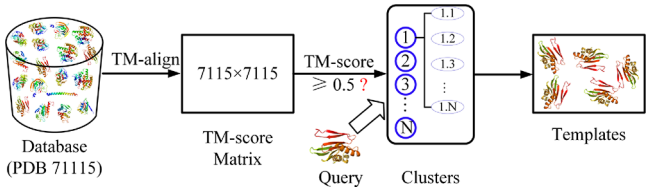


Figure 1. Architecture of TM-search for progressively searching the similar structures.

MATERIALS AND METHODS

Preparation of the TM-search Database. In this study, we construct a hierarchical structure database for the TM-search according to the following procedure. First, each structure in the database was separated into domains. If the domains of a protein are defined by the SCOPe (Structural Classification of Proteins - extended),[23] then the SCOPe definition is used for domain splitting. Otherwise, the structure is split into domains using the Protein Domain Parser (PDP) program.[24] After excluding structures with less than 30 residues, we get ∼470,000 structures in the initial database (PDBall). Subsequently, we used CD-HIT to cluster the database at 70% sequence identity. The biggest structure from each cluster is used to represent the cluster, and it generates 71,115 nonredundant databases (PDB70).

Next, we used TM-align to calculate the TM-score between each pair of structures in PDB70 to generate a TM-score similarity matrix, based on which we group the PDB70 into structurally similar clusters. The clusters were built in-house using agglomerate hierarchical clustering algorithms (Figure 2). During clustering, the first step is to sort proteins in descending order of length and identify the representative structure with the maximum sequence length. All structures with a TM-score ≥0.5 to this representative will then be grouped into the same cluster and excluded from the TM-score matrix. Then, the next representative and the corresponding members are obtained from the remaining proteins. These iterations are repeated until no unclustered structure remains.

Since the structure search in the TM-search begins with alignment through the cluster representative, the selection of the representative is a critical factor influencing the speed and accuracy of TM-search. We implement three representative selection strategies: type-α, type-$\alpha\beta$, and type-β. In the type-α strategy, cluster representative is the protein with the maximum number of neighbors with a TM-score ≥0.5. If two or more proteins have the maximum number of neighbors, then one of them will be arbitrarily chosen as the representative (Figure 2A). The type-$\alpha\beta$ strategy is almost

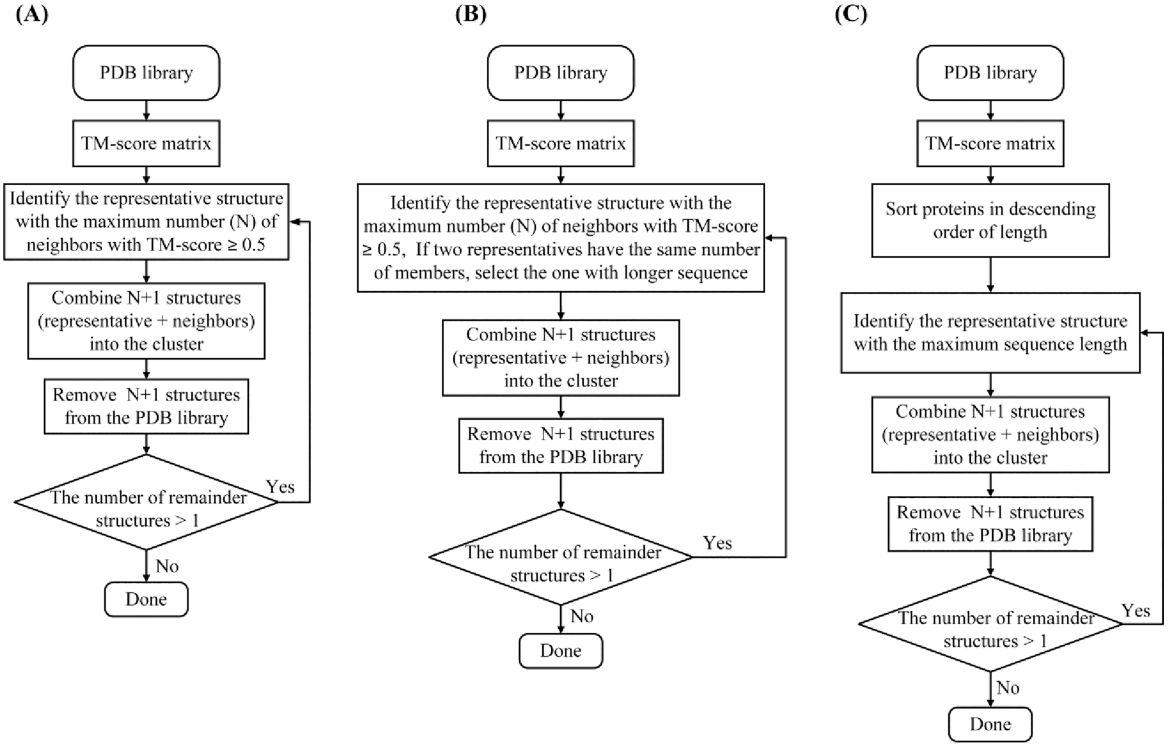


Figure 2. Flowchart of three clustering algorithms for TM-search including (A) type-α, (B) type-$\alpha\beta$, and (C) type-β.

https://doi.org/10.1021/acs.jcim.3c01455
J. Chem. Inf. Model. 2024, 64, 1043−1049

identical to type-α, except that if two or more proteins all have the maximum number of neighbors, then the protein with the longest sequence will be chosen as the representative (Figure 2B). On the other hand, in the type-β strategy, the longest protein not belonging to an existing cluster is chosen as the representative (Figure 2C).

The database is updated each week in accordance with the PDB. Briefly, a new structure released by the PDB will be compared to all of the existing representatives by TM-align. The new protein chains will then be added to the cluster whose representative has the highest TM-score if the TM-score ≥0.5. Otherwise, the new structure will become representative of a new cluster.

Evaluation Metric. To evaluate the performance of the TM-search, we have used two well-known metrics: precision,[25] recall,[25] F_1-score,[25] and AUROC.[26] The precision (P) and recall (R) are defined as

$$P(n) = \frac{\mathrm{TP}(n)}{n} \tag{1}$$

and

$$R(n) = \frac{\mathrm{TP}(n)}{T} \tag{2}$$

where n is the total number of top hits in the structure search result list for the query protein, $\mathrm{TP}(n)$ is the number of true positive (TM-score ≥0.5 between the query and the template hit) among the first n pairs in the ordered list of results, and T is the number of similar structures with a TM-score ≥0.5 to the query in the database. The precision and recall are undefined if the top hits n are larger than the size of the result list M. The values of imputed precision (P') and recall (R') are defined as

$$P'(n) = \begin{cases} \dfrac{\mathrm{TP}(n)'}{n}, & \text{if } n \le M \\ \dfrac{\mathrm{TP}(M)'}{M}, & \text{if } n > M \end{cases} \tag{3}$$

and

$$R'(n) = \begin{cases} \dfrac{\mathrm{TP}(n)'}{T}, & \text{if } n \le M \\ \dfrac{\mathrm{TP}(M)'}{T}, & \text{if } n > M \end{cases} \tag{4}$$

where $\mathrm{TP}(n)' \le T$. The F_1-score is the harmonic mean of the precision and recall. The F_1-score (F_1) is defined as

$$F_1(n) = 2\cdot\frac{P(n)R(n)}{P(n)+R(n)} = \frac{2\mathrm{TP}(n)}{n+T} \tag{5}$$

The AUROC is the area under the curve (AUC) of the cumulative ROC curve. The AUROC is used to evaluate the accuracy of neighbor protein structure (similar protein structure) retrieval and has been widely used in many research areas, including the protein structure alignment area.[11,26]

Algorithm for Database Search. TM-search uses the highly efficient pairwise protein structure alignment program TM-align, which uses TM-score as the objective function. The TM-score is the length-independent scoring function for measuring the similarity of two structures:

$$\mathrm{TM-score} = \max \frac{1}{L}\sum_{i}^{L_{\mathrm{ali}}} \frac{1}{1+\left(\frac{d_i}{d_0}\right)^2} \tag{6}$$

Here, L is the query protein length, L_{ali} is the number of aligned residues, and d_i is the distance between the ith pair of C_α atoms of the aligned residue pair. d_0 is a scale factor that depends on the size of protein length, defined as

$$d_0 = \begin{cases} 1.24\cdot\sqrt[3]{L-15} - 1.8 & \text{if } L > 21 \\ 0.5 & \text{otherwise} \end{cases} \tag{7}$$

The range of TM-score is 0−1, where TM-score ≥0.5 indicates the pair of structures sharing the same overall topology.[27]

Given a query protein structure to search the database, first, the query structure is compared with all representative structures of each cluster. Then, all the representative structures with a TM-score <0.5 will be discarded, and the corresponding members will not be aligned against the query. Otherwise, after finding the clusters with representatives that match well against the query structure, each of the remaining members of the clusters is aligned with the query. This approach avoids performing the time-consuming alignment for dissimilar structure pairs.

RESULTS AND DISCUSSION

Benchmark Databases. To evaluate various representative selection strategies for constructing the TM-search database, we randomly selected a set of 200 nonhomologous protein chains, referred to as PDB200, from the Protein Data Bank (PDB), which range in size from 46 to 1058 residues and whose pairwise sequence identity is <30%. Furthermore, to compare the performance of our method with the existing ones, we collected another data set called PDB1200, consisting of 1200 protein chains not included in the TM-search database, also ranging in size from 30 to 781 residues. The complete benchmark data sets can be found in the PDB1200

Table 1. Comparison of the Three Clustering Algorithm Types (Type-α, Type-$\alpha\beta$, and Type-β) Using the PDB200 Dataset

		average numbers of similar structures			
type	representative numbers	TM-score ≥ 0.5	TM-score ≥ 0.6	TM-score ≥ 0.7	average time (s)
type-α	4018	339.16	86.82	30.08	813.97
type-$\alpha\beta$	4016	339.15	86.82	30.08	804.35
type-β	3834	343.29	87.34	30.15	768.10

Table 2. Comparison of Methods on the PDB1200 Dataset

		average numbers of similar structures		
methods	average time (s)	TM-score ≥ 0.5	TM-score ≥ 0.6	TM-score ≥ 0.7
PSI-BLAST	43.04	41.16	41.16	28.71
Foldseek	22.90	74.65	53.75	33.69
Dali	9859.27	301.12	112.65	43.71
TM-align	4834.25	875.32	218.47	66.63
TM-search	174.81	758.09	202.05	61.70

https://doi.org/10.1021/acs.jcim.3c01455
J. Chem. Inf. Model. 2024, 64, 1043−1049

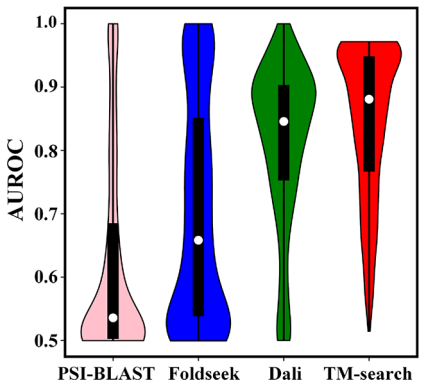


Figure 3. Distribution of AUROC for detection of analogous templates with TM-score ≥0.5 by four methods. TM-search achieves the highest average AUROC of 0.846

benchmark distribution package (https://zhanggroup.org/TM-search/ benchmark/).

In addition, we constructed a benchmark data set (SCOPe350) consisting of 200 randomly selected domains from the SCOPe40 data set.[28] This data set is the full set of all 15,176 SCOPe domains from the ASTRAL[29] subset (i.e., nonredundant set with <40% sequence identity) of SCOPe database version 2.08.

Impact of Representative Selection on TM-search Performance. When benchmarked on the PDB200 data set, the three strategies do not result in noticeable differences in sensitivities, as measured by the number of hits with TM-scores ≥0.5, ≥0.6, and ≥0.7 (Table 1). On the other hand, the speed of TM-search is highest when using type-β, mainly because it has the smallest number of clusters. Therefore, we use type-β as the strategy for cluster representative selection in TM-search.

Overall Performance Evaluations. We demonstrated the overall performance of TM-search to search structure databases for structurally analogous proteins in comparison to four existing search programs: PSI-BLAST,[30] Dali,[25] Foldseek,[8] and TM-align.[7] These programs were installed locally and ran with the default parameters. The setting is as follows: (i) DALI: import.pl --pdbfile query.pdb --pdbid PDBid --dat./DAT/; dali.pl --cd1 queryID (DAT id) --db DB.list -TITLE systematic --dat1 DAT/ --dat2 DAT/ --outfmt summary --clean; (ii) Foldseek: foldseek createdb library/libDB; foldseek easy-search query.pdb libDB outputfile.a3m -s 9.5 tmp1; (iii) PSI-BLAST: psiblast -query query.fasta -db pdb.fasta -evalue 10 -num_iterations 3 -out output.psi -outfmt 6. To ensure an equitable comparison, we used the specificity scores derived from the various comparison methods as the selection criteria. Here, although the PSI-BLAST search is based on sequence rather than structure, we include it in our benchmark anyway to show the gap in analogous structure detection between sequence-based and structure-based approaches. The TM-align search result is considered the ground truth. The comparison uses query structures from the PDB1200 benchmark data sets.

Table 2 shows the average number of similar structures identified under three different TM-score cutoffs (TM-scores ≥0.5, ≥0.6, and ≥0.7) and the average computing time. For all

Table 3. Evaluations of the Top 150 Predictions

method	F_1-score	precision	recall
TM-search	0.46	0.70	0.57
Dali	0.37	0.59	0.45
Foldseek	0.35	0.60	0.38
PSI-BLAST	0.23	0.46	0.22

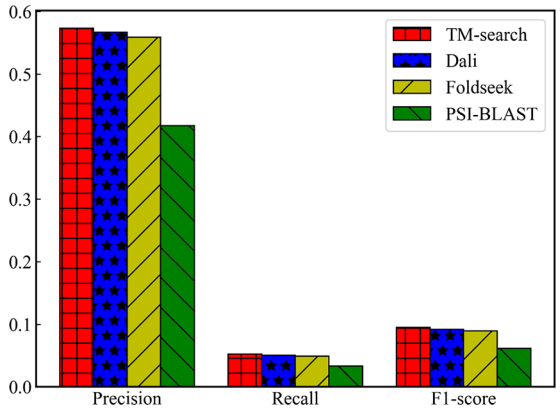


Figure 5. Comparison of the average precision, recall, and F_1-score of TM-search, Dali, Foldseek, and PSI-BLAST on the SCOPe350 database.

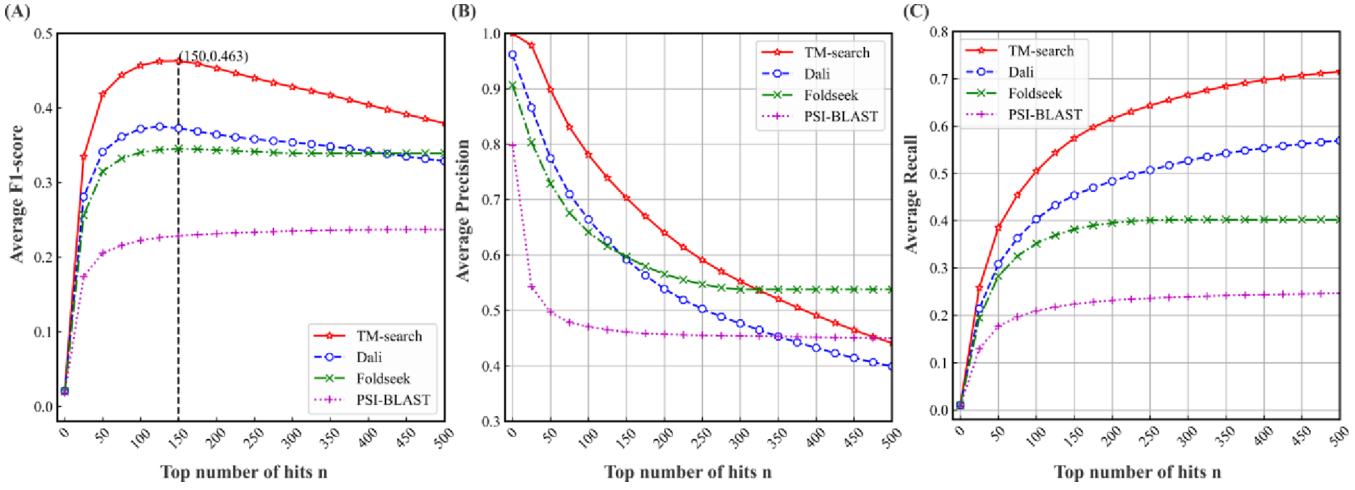


Figure 4. Performance of the database search by TM-search, Dali, Foldseek, and PSI-BLAST, measured by the average (A) F_1-score, (B) precision, and (C) recall for the top n template hits.

https://doi.org/10.1021/acs.jcim.3c01455
J. Chem. Inf. Model. 2024, 64, 1043−1049

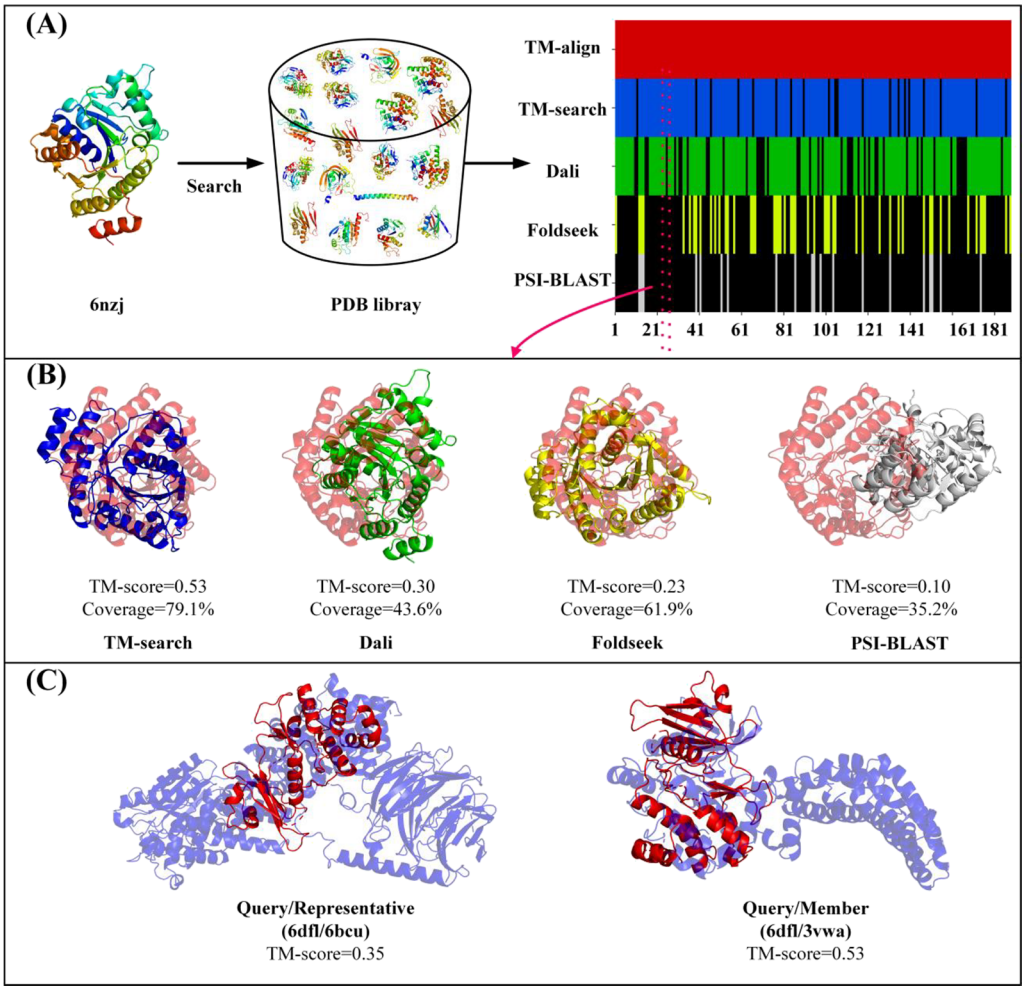


Figure 6. Structure database search for nitrogenase iron protein (PDB code: 6nzj chain A). (A) Performance overview: The heatmap on the right displays 184 structures with a TM-score ≥0.5, sorted by TM-align. Colored regions show hits and misses by TM-search (blue), Dali (green), Foldseek (yellow), and PSI-BLAST (gray), identifying 174, 146, 61, and 21 structures, respectively. (B) Structure alignment example: Query protein (opaque) aligned with a structure (PDB: 1rh9 chain A, semitransparent) found only by TM-search. (C) Superimposed structures: Query structure (PDB: 6dfl chain A, opaque red) superimposed on the representative (PDB: 6bcu chain W) and a member (PDB: 3vwa chain A) in semitransparent blue.

three TM-score cutoffs, TM-search consistently identifies a much larger number of similar structures than the PSI-BLAST, Dali, and Foldseek and performs similarly to the ground-truth TM-align in terms of sensitivity. The difference is particularly pronounced for a TM-score cutoff of 0.5, where the average number of similar structures found by TM-search (758.09) is 18.1, 2.5, and 10.2 times larger than PSI-BLAST, Dali, and Foldseek, respectively.

AUROC can describe the sorting capability of a search tool.[11] Figure 3 shows the average AUROC across the benchmark PDB1200 database. TM-search's average AUROC is 0.846, which is 4.0, 20.7, and 36.7% higher than Dali, Foldseek, and PSI-BLAST, respectively. Therefore, TM-search is more effective in sorting similar versus dissimilar structures compared to these three existing programs.

Furthermore, we compared the search speed of the programs defined as the average running time (in seconds) used for each query structure in the benchmark PDB1200. All of the programs were executed on the Linux system with an Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20 GHz CPUs. TM-search takes ∼3 min per query structure, which is 26.7 and 53.3 times faster than TM-align and Dali, respectively. Although TM-search is slower than PSI-BLAST and Foldseek, its significantly higher sensitivity still makes it a useful tool when the complete and accurate identification of hits is desired.

The precision of the top prediction is another index to evaluate the effectiveness of search engines. Specifically, the principle is to list as many true positive results as possible on the top of the list, which can be evaluated by the precision of top predictions. Figure 4A−C presents plots for F_1-score, precision, and recall for the first n = 1, 2, 3,..., 500 results, respectively, for all compared methods. When evaluating the top 150 structures, TM-search exhibits a better F_1-score compared to those of Dali, Foldseek, and PSI-BLAST. The precision and recall for the TM-search are the highest among all compared methods at all cutoffs (Figure 4B,C). Table 3 lists the evaluation results for the top 150 predictions. Compared with Dali, the TM-search promotes the average F_1-score by 24.3%. In addition, we calculate the average precision and recall, and the TM-search values reach 0.70 and 0.57, respectively, whereas the other methods achieve at most 0.60 and 0.45.

Evaluation of Database Searches Based on the SCOPe Reference Fold. We evaluated the precision and

https://doi.org/10.1021/acs.jcim.3c01455
J. Chem. Inf. Model. 2024, 64, 1043−1049

recall of the TM-search and four protein domain comparison tools using the SCOPe databases, which rely on the fold definition in SCOPe. A structure is considered a true positive (TP) (eq 1) if the fold definition of the query matches that of the first n structures in the result list. T (eq 2) is the number of structures in the fold class.

Figure 5 compares the average precision, recall, and F_1-score on the SCOPe350 target protein. From Figure 5, it is easily found that TM-search consistently outperforms the other three control methods concerning the three evaluation indices. Concretely, TM-search gains 3.4, 5.8, and 50.3% average improvements in the above-mentioned three evaluation metrics compared to Dali, Foldseek, and PSI-BLAST, respectively. The comparison results presented above show that TM-search is better than Dali, Foldseek, and PSI-BLAST on the SCOPe350 database.

As a case study, we show in Figure 6 the structure database search result for the top 184 hits of PDB 6nzj. As shown in Figure 6A, TM-search identifies 174 hits with a TM-score >0.5, which is 1.4, 2.9, and 8.3 times more than Dali, Foldseek, and PSI-BLAST, respectively. Figure 6B shows the superposed structure between the query and the high TM-score hits identified (PDB code: 1rh9) by TM-search but missed by Dali, Foldseek, and PSI-BLAST. PSI-BLAST missed this hit probably due to low sequence similarity (sequence identity = 19.8%). Dali and Foldseek also failed to generate a good alignment for this hit, resulting in low coverages of 43.4 and 61.9% and low TM-scores of 0.30 and 0.23, respectively, which are much lower than those achieved by TM-search (TM-score = 0.53, coverage = 79.1%). In Figure 6C, we show another example for the WaaP lipopolysaccharide heptose kinase (PDB code: 6dfl), where TM-search failed to identify a hit (PDB code: 3vwa) with a TM-score of 0.53, and it was attributed to the TM-score of the query protein with the representative structure of the reference structure (PDB code: 6bcu) being less than 0.5 (TM-score = 0.35); thus, the query will not continue the search member of the representative, and the structure alignments were obtained using TM-align.

CONCLUSIONS

In this work, we developed TM-search, an efficient and effective algorithm to retrieve similar protein structures from the PDB. Large-scale benchmarks show that TM-search balances search speed and sensitivity well compared to existing programs, including Dali, Foldseek, PSI-BLAST, and TM-align.

One limitation of the TM-search database preparation is that new structures from weekly releases of the PDB can only be added as new clusters or members of old clusters without changing existing cluster representatives. Future work will focus on the development of a more effective strategy to update clusters in the TM-search database.

ASSOCIATED CONTENT

Data Availability Statement

The data set used in this study and source code are freely available at https://zhanggroup.org/TM-search/.

AUTHOR INFORMATION

Corresponding Authors

Yang Zhang − *Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, Michigan 48109-2218, United States;* orcid.org/0000-0002-2739-1916; Email: zhang@zhanggroup.org

Dong-Jun Yu − *School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China;* orcid.org/0000-0002-6786-8053; Email: njyudj@njust.edu.cn

Authors

Zi Liu − *School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China; Computer Department, Jingdezhen Ceramic University, Jingdezhen 333403, China*

Chengxin Zhang − *Department of Computational Medicine and Bioinformatics, University of Michigan, Ann Arbor, Michigan 48109-2218, United States*

Qidi Zhang − *Computer Department, Jingdezhen Ceramic University, Jingdezhen 333403, China*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jcim.3c01455

Author Contributions

#Z.L. and C.Z. contributed equally to this work.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China (62372234, 62072243, 61772273, and 62076111) and the Natural Science Foundation of Jiangsu (BK20201304).

REFERENCES

(1) Laskowski, R. A.; Watson, J. D.; Thornton, J. M. ProFunc: a server for predicting protein function from 3D structure. *Nucleic Acids Res.* **2005**, *33*, W89−W93.

(2) Zhang, C. X.; Freddolino, P. L.; Zhang, Y. COFACTOR: improved protein function prediction by combining structure, sequence and protein-protein interaction information. *Nucleic Acids Res.* **2017**, *45*, W291−W299.

(3) Zhang, J.; Liang, Y.; Zhang, Y. Atomic-level protein structure refinement using fragment-guided molecular dynamics conformation sampling. *Structure* **2011**, *19*, 1784−95.

(4) Pearce, R.; Huang, X. Q.; Setiawan, D.; Zhang, Y. EvoDesign: Designing Protein-Protein Binding Interactions Using Evolutionary Interface Profiles in Conjunction with an Optimized Physical Energy Function. *J. Mol. Biol.* **2019**, *431*, 2467−2476.

(5) Burley, S. K.; Berman, H. M.; Kleywegt, G. J.; Markley, J. L.; Nakamura, H.; Velankar, S. Protein Data Bank (PDB): The Single Global Macromolecular Structure Archive. *Methods Mol. Biol.* **2017**, *1607*, 627−641.

(6) Varadi, M.; Anyango, S.; Deshpande, M.; Nair, S.; Natassia, C.; Yordanova, G.; Yuan, D.; Stroe, O.; Wood, G.; Laydon, A.; Zidek, A.; Green, T.; Tunyasuvunakool, K.; Petersen, S.; Jumper, J.; Clancy, E.; Green, R.; Vora, A.; Lutfi, M.; Figurnov, M.; Cowie, A.; Hobbs, N.; Kohli, P.; Kleywegt, G.; Birney, E.; Hassabis, D.; Velankar, S. AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Res.* **2022**, *50*, D439−D444.

(7) Zhang, Y.; Skolnick, J. TM-align: a protein structure alignment algorithm based on the TM-score. *Nucleic Acids Res.* **2005**, *33*, 2302−9.

(8) van Kempen, M.; Kim, S. S.; Tumescheit, C.; Mirdita, M.; Söding, J.; Steinegger, M. Foldseek: fast and accurate protein structure search. *bioRxiv* **2022**, DOI: 10.1101/2022.02.07.479398.

(9) van Kempen, M.; Kim, S. S.; Tumescheit, C.; Mirdita, M.; Lee, J.; Gilchrist, C. L. M.; Söding, J.; Steinegger, M. Fast and accurate protein structure search with Foldseek. *Nat. Biotechnol.* **2023**, DOI: 10.1038/s41587-023-01773-0.

(10) Liu, Y.; Ye, Q.; Wang, L. W.; Peng, J. Learning structural motif representations for efficient protein structure search. *Bioinformatics* **2018**, *34*, 773−780.
(11) Min, Y. S.; Liu, S.; Lou, C. Y.; Cui, X. F. Learning Protein Structural Fingerprints under the Label-Free Supervision of Domain Knowledge. *IEEE Int. Conf. Bioinf. Biomed.* **2018**, 69−74.
(12) Aderinwale, T.; Bharadwaj, V.; Christoffer, C.; Terashi, G.; Zhang, Z.; Jahandideh, R.; Kagaya, Y.; Kihara, D. Real-time structure search and structure classification for AlphaFold protein models. *Commun. Biol.* **2022**, *5*, 316.
(13) Xia, C.; Feng, S.-H.; Xia, Y.; Pan, X.; Shen, H.-B. Fast protein structure comparison through effective representation learning with contrastive graph neural networks. *PLoS Comput. Biol.* **2022**, *18*, No. e1009986.
(14) Deng, L.; Zhong, G.; Liu, C.; Luo, J.; Liu, H. MADOKA: an ultra-fast approach for large-scale protein structure similarity searching. *BMC Bioinf.* **2019**, *20*, 662.
(15) Malod-Dognin, N.; Pržulj, N. GR-Align: fast and flexible alignment of protein 3D structures using graphlet degree similarity. *Bioinformatics* **2014**, *30*, 1259−1265.
(16) Holm, L.; Rosenstrom, P. Dali server: conservation mapping in 3D. *Nucleic Acids Res.* **2010**, *38*, W545−W549.
(17) Ye, Y. Z.; Godzik, A. Flexible structure alignment by chaining aligned fragment pairs allowing twists. *Bioinformatics* **2003**, *19*, ii246−ii255.
(18) Mirdita, M.; Steinegger, M.; Söding, J. MMseqs2 desktop and local web server app for fast, interactive sequence searches. *Bioinformatics* **2019**, *35*, 2856−2858.
(19) Huang, Y.; Niu, B.; Gao, Y.; Fu, L.; Li, W. CD-HIT Suite: a web server for clustering and comparing biological sequences. *Bioinformatics* **2010**, *26*, 680−682.
(20) Dong, R. Z.; Peng, Z. L.; Zhang, Y.; Yang, J. Y. mTM-align: an algorithm for fast and accurate multiple protein structure alignment. *Bioinformatics* **2018**, *34*, 1719−1725.
(21) Mezulis, S.; Sternberg, M. J. E.; Kelley, L. A. PhyreStorm: A Web Server for Fast Structural Searches Against the PDB. *J. Mol. Biol.* **2016**, *428*, 702−708.
(22) Zhang, Y.; Skolnick, J. Scoring function for automated assessment of protein structure template quality. *Proteins-Structure Function and Bioinformatics* **2004**, *57*, 702−710.
(23) Hubbard, T.; Murzin, A. G.; Brenner, S. E.; Cyrus, C. SCOP: A Structural Classification of Proteins database. *Nucleic Acids Res.* **1997**, 236−239.
(24) Alexandrov, N.; Shindyalov, I. PDP: protein domain parser. *Bioinformatics* **2003**, *19*, 429−430.
(25) Holm, L. Benchmarking fold detection by DaliLite v.5. *Bioinformatics* **2019**, *35*, 5326−5327.
(26) Cui, X. F.; Li, S. C.; He, L.; Li, M. Fingerprinting protein structures effectively and efficiently. *Bioinformatics* **2014**, *30*, 949−955.
(27) Jinrui, X.; Yang, Z. How significant is a protein structure similarity with TM-score = 0.5? *Bioinformatics* **2010**, 889−95.
(28) Fox, N. K.; Brenner, S. E.; Chandonia, J.-M. SCOPe: Structural Classification of Proteins—extended, integrating SCOP and ASTRAL data and classification of new structures. *Nucleic acids research* **2014**, *42*, D304−D309.
(29) Brenner, S. E.; Koehl, P.; Levitt, M. The ASTRAL compendium for protein structure and sequence analysis. *Nucleic acids research* **2000**, *28*, 254−256.
(30) Altschul, S. F.; Madden, T. L.; Schaffer, A. A.; Zhang, J.; Zhang, Z.; Miller, W.; Lipman, D. J. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.* **1997**, *25*, 3389−402.

https://doi.org/10.1021/acs.jcim.3c01455
J. Chem. Inf. Model. 2024, 64, 1043−1049