Exploiting Structural Constraints of Proteolytic Catalytic Triads for Fast Supercomputer Scaffold Probing in Enzyme Design Studies

Alexander Zlobin[1,2,3(✉)], Alexander-Pavel Ermidis[3], Valentina Maslova[2,3], Julia Belyaeva[2,3], and Andrey Golovin[1,2,3(✉)]

[1] Shemyakin–Ovchinnikov Institute of Bioorganic Chemistry, Russian Academy of Sciences, 117997 Moscow, Russian Federation
zlobin.as@talantiuspeh.ru, golovin@belozersky.msu.ru
[2] Sirius University of Science and Technology, 354340 Sochi, Russian Federation
[3] Faculty of Bioengineering and Bioinformatics, Lomonosov Moscow State University, 119991 Moscow, Russian Federation
{alexandrpavele, val_ma, belyaevajuly}@fbb.msu.ru

Abstract. Evolutionary constraints on the effectiveness of enzymatic function result in well-defined architectures of active sites. In this study we show that these constraints are fully pronounced even at the backbone level. We explore the possibility of defining catalytic triads in proteases just by their relative backbone orientations to dramatically speed up the scaffold search problem of *de novo* enzyme design. An order of magnitude speed-up achieved this way paves a way to a routine scanning of the whole structural proteome including modeled structures.

Keywords: Enzyme design · Structural bioinformatics · Structural similarity · Protease

1 Introduction

Proteases comprise a group of structurally and functionally diverse enzymes that have the common ability to catalyze the hydrolysis of peptide bonds [1]. This ability is facilitated by active sites of varying composition that give different classes of proteolytic enzymes their respective names. This way there are serine, cysteine, threonine, aspartyl, glutamyl proteases and metalloproteases. Among these groups, serine and cysteine proteases are the most studied and act under the widest condition ranges. Most serine proteases employ a catalytic triad consisting of Ser, His and Asp residues as an active site. Three residues must be positioned in a specific way to facilitate Ser deprotonation in order to perform a nucleophilic attack on a substrate (Fig. 1). Ser must be hydrogen bonded to His, which in turn has a second hydrogen bond to Asp that positions His correctly and shifts its pKa. More generally and according to the

V. Voevodin and S. Sobolev (Eds.): RuSCDays 2021, CCIS 1510, pp. 58–72, 2021.
https://doi.org/10.1007/978-3-030-92864-3_5

performed function these three residues are called a nucleophile, a base and an acid (or an activator). A base formed by histidine is a widespread scenario. In turn, a nucleophile and an activator may be different from the most common Ser and Asp residues [2]. Thereby exist classes of triad-harbouring cysteine proteases [3]. In general, this evolutionary successful arrangement of three residues is exploited for other hydrolytic functions. Despite the pronounced need for a specific hydrogen bonding of triad residues, it can be achieved by more than one spatial arrangement. Comparisons of single representative enzymes showed that, for example, inside the class of serine triad-harbouring proteases, chymotrypsin and subtilisin implement different triad architectures. On the other hand, cysteine TEV protease is remarkably similar to serine protease trypsin in terms of active site spatial arrangement. Systematic investigation into the space of all triad architectures was not performed to date.

Animal, plant, and, especially, microbial proteases represent the largest and most important segment of an industrial enzyme market where they are used in detergents, food processing or leather industry, as biocatalysts in organic synthesis, and as therapeutics. The list of potential practical applications of proteases can be greatly expanded, especially for therapeutic applications, once their catalytic activities can be engineered for specific uses [1]. This can be done with the help of state-of-the-art computational techniques. A combination of structural analysis, reaction modelling and rational design can be used to modify specificity, stability or other properties of existing enzymes, including proteases [4–7]. Among the examples of sound success stories in protease design is Kuma062, a kumamolisin variant repurposed to process gluten [8]. To face humanity's demand for applied proteolytic functions, modification of existing enzymes alone is, however, not sufficient. The computational methods of *de novo* introduction of enzymatic function into previously non-catalytic protein folds are regarded as a major step forward in addressing the needs of industry and medicine [9]. These efforts are, however, limited to the approach implemented in Rosetta3 enzyme design protocol [10]. This method was previously successfully used for transfer of existing active sites into manually generated folds and for computational design of previously non-existent enzymatic functions such as retro-aldolase or Diels-Alder reaction catalysts [11–14].

The underlying computational procedure starts from the definition of a theozyme – a set of atoms at their respective coordinates mimicking the crucial step of the enzymatic process, e.g. transition state. Most commonly a theozyme is constructed from the substrate moiety and the sidechains of active site residues. After theozyme is constructed, a suitable backbone scaffold to harbour its residues needs to be obtained either by searching the space of known structures or by constructing it from scratch. One can focus only on backbone scaffolds because there is only a limited set of them, and they are highly degenerate in terms of underlying sequences; thus it is unnecessary to perform placement search in all individual proteins with known structure. The searching algorithm implemented in the Rosetta Match application is rather slow as it scans through rotameric libraries of the desired active site residues' sidechains. The sampling takes even longer if the geometry constraints tolerate fluctuations in the theozyme structure. The further design is based on preservation and additional stabilisation of some interactions between the active site sidechains and the transitional state of the reaction [10].

Such technique does not require the similarity between the active site backbone conformations in the newly constructed enzyme and in the initial source of theozyme, and relies on idealized backbone-dependent rotamer libraries for sidechains. However, in some enzymes the backbone of the active site residues plays a key role in the oxyanion hole formation. For example, both in serine and cysteine proteases the catalytic Ser/Cys backbone N forms a hydrogen bond with the carbonyl O of the substrate. Such interaction is crucial for the catalysis [15, 16]. Moreover, it was shown that the residues directly involved in the catalytic act more often are rotameric outliers [17, 18].

Conformations of protease catalytic residues are highly specific being the result of evolutionary selection. Since rotamer distributions are inherently backbone conformation-dependent, and since backbone is likely to itself participate in the reaction, we suggest that relative backbone geometries of catalytic residues are themselves highly specific. What is more, we hypothesize that by knowing the relative geometries of the triad's backbones one can derive the triad's full structure. It makes it possible to make theozyme placement search task completely sidechain-agnostic. In this work we provide justification for this idea. We propose a description of the catalytic site using the involved residues' backbone orientation. We demonstrate a distinguishable difference between triads in active sites found in available serine and cysteine proteases and other non-catalytic combinations of the same residues. Once the hypothesis is proven, we show that the natural consequence of it is the possibility to drastically speed up the scaffold searching. We present a computational protocol for theozyme placement based on scaffold backbone orientation analysis. When applied to the search for trypsin catalytic triad placement, our backbone-based approach outperforms Rosetta Match in speed by at least 30 times while retaining the accuracy. Low computational cost of the presented solution allows one to run over about 180 000 structures (a full PDB) placement search for one active site in a matter of minutes when using supercomputer resources.

2 Materials and Methods

2.1 Backbone-Based Vectorization of Triads

We define triad as a triplet of unique protein residues with known position of their backbone atoms N, CA, C. For each residue we introduce a virtual point in space called BB placed at the geometric center between its N and C atoms. For a pair of residues i and j, a number of terms is computed. Term α_{ij} is defined as an angle between atoms i_C–i_{BB}–j_{BB}. Term θ_{ij} is defined as a torsion angle constructed for atoms i_C–i_{BB}–j_{BB}–j_C. Term η_{ij} is defined as a torsion angle constructed for atoms i_{CA}–i_C–i_{BB}–j_{BB}. Triad vector $\boldsymbol{V}$ for residues i, j, k is then constructed from these terms as follows:

$$\boldsymbol{V}_{ijk} = \{\alpha_{ij}, \alpha_{ji}, \theta_{ij}, \eta_{ij}, \alpha_{jk}, \alpha_{kj}, \theta_{jk}, \eta_{jk}, \alpha_{ki}, \alpha_{ik}, \theta_{ki}, \eta_{ki}\} \quad (1)$$

Throughout the paper all angular terms are expressed in degrees.

2.2 Protease Triads Dataset Construction

For this work, a collection of PDB IDs matching EC codes 3.4.21 (Serine endopeptidases) and 3.4.22 (Cysteine endopeptidases) with resolution under 3 Å and R-free under 0.4 was obtained. To ensure non-redundancy the dataset was culled at the 90% sequence similarity level using Pisces [19]. The resulting PDB IDs dataset comprised 811 entries.

We then searched for catalytic-like triads in the structures of these proteins. Search and analysis was performed with the help of ProDy [20]. First, all histidine residues were selected. We then analyzed the surroundings of both its sidechain N_δ and N_ε nitrogen atoms. Catalytic-like triad was identified as a triplet of residues Nuc-His-Act, where Nuc (nucleophile) is either Ser or Cys and Act (activator) is either Asp, Glu, Asn or Gln, if there was simultaneously Oγ or Sγ atom of Nuc closer then 3.5 Å to any one nitrogen atom of His and one of the sidechain oxygens of Act closer then 3.5 Å to another nitrogen of His. If the analyzed structure comprised several copies of the same subunit the triad from only one of them was retained for subsequent studies.

For each catalytic-like triad obtained this way a triad vector was computed as described above. The resulting triad dataset comprised 312 entries.

2.3 Clusterization of Triad Vectors

We chose to compose our vector only from angular and torsional terms to avoid normalization problems since all the values are expressed in the same units and lie in the same range. However, half of vector values represent torsions which are naturally periodic. Because of this, straightforward implementation of distance-based clusterization is incorrect since commonly used distance metrics are not periodic. We thus precompute the distance matrix manually. For two triad vectors $\boldsymbol{V_{ijk}}$ and $\boldsymbol{V_{abc}}$, the distance D between them is the Euclidean norm of a vector $\boldsymbol{\Delta V}$:

$$D = \|\Delta \boldsymbol{V}\|_2 = \left\|\{\Delta\alpha_{ij,ab}, \Delta\alpha_{ji,ba}, \Delta\theta_{ij,ab}, \Delta\eta_{ij,ab}, \Delta\alpha_{jk,bc}, \Delta\alpha_{kj,cb}, \Delta\theta_{jk,bc}, \Delta\eta_{jk,bc}, \Delta\alpha_{ki,ca}, \Delta\alpha_{ik,ac}, \Delta\theta_{ki,ca}, \Delta\eta_{ki,ca}\}\right\|_2 \quad (2)$$

Where

$$\Delta\alpha_{ij,ab} = \alpha_{ij} - \alpha_{ab} \quad (3)$$

$$\Delta\theta_{ij,ab} = min\left(\left|\theta_{ij} - \theta_{ab}\right|, 360 - \left|\theta_{ij} - \theta_{ab}\right|\right) \quad (4)$$

$$\Delta\eta_{ij,ab} = min\left(\left|\eta_{ij} - \eta_{ab}\right|, 360 - \left|\eta_{ij} - \eta_{ab}\right|\right) \quad (5)$$

and similar for all other instances of α, θ and η.

Precomputed distance matrix was utilized to perform density-based clusterization. DBScan from the sklearn Python package was utilized for the task [21]. The epsilon parameter, specifying the maximum distance between two samples for one to be considered as in the neighborhood of the other, was set to 50. The number of samples

in a neighborhood for a point to be considered as a core point was set to 10. 4 clusters were identified, with 86 points not being in any of them.

2.4 Visualization of Clusterization Results

Informative visualization of clusterization results of data represented as a 12-dimensional vector naturally calls for a dimensionality reduction. UMAP technique was selected for the task, implemented in the umap-learn Python package [22]. All parameters were set to default except for the metric which in our case was set to "precomputed" since we used an already built distance matrix as an input.

Protein structures with triads from the same clusters were superposed with the help of pair_fit functionality in PyMol, which was also used for molecular visualization throughout the paper [23].

2.5 Scaffold Preprocessing and Placement Search Procedure

For a given protein scaffold query and a triad vector template the placement search procedure is intended to produce a ranked list of triples of scaffold residues most closely matching the relative backbone orientation of the template. To enforce reusability, a protein scaffold is first preprocessed. Protein structure is transformed into a graph with its residues represented as nodes in this graph. The edge between two nodes i and j is drawn if the distance between CA atoms of two respective residues (d_{CA}) lies between 4 and 13 Å and the distance between their CB atoms (d_{CB}) does not exceed d_{CA} by more then 1 Å. The edge is assigned a data container comprised of two vectors $\{\alpha_{ij}, \alpha_{ji}, \theta_{ij}, \eta_{ij}\}$ and $\{\alpha_{ji}, \alpha_{ij}, \theta_{ji}, \eta_{ji}\}$. The list of all triads is then obtained by performing a clique search and selecting all the cliques of length 3. An additional filter is imposed on a triad ijk so that the area of the triangle with vertices CA_i, CA_j and CA_k does not exceed 35 $Å^2$. As discussed earlier, the construction of a final triad vector relies on specifying the sequence of its constitutive residues. For all selected cliques all permutations of its vertices are constructed and assigned a triad vector by combining the respective components of a data stored on the graph's edges. The final list of all triad vectors and respective residue indices in each sequential order is saved as a Python pickle for further use. All graph manipulations are performed with the help of the networkx Python package [24].

For a placement search for a specified template and a scaffold a list of stored triad vectors is further reduced by considering that only half of the six permutations of triad indices are relevant for each single search task. For the input template triad ijk it is calculated whether the triple of vectors $\underline{N_iC_i}$, $\underline{N_jC_j}$, $\underline{N_kC_k}$ is right-handed or otherwise, and only matching triples from the scaffold are retained for search. Finally, distance between each scaffold triad vector and template vector is calculated as described earlier, and scaffold positions are reported if such distance is below the threshold.

2.6 Scaffold Library Construction

CATH non-redundant S40 collection of domains was obtained as PDB files totaling 31879 scaffolds [25]. Since the position of CB atoms is a prerequisite for one of the filter stages in a placement search, all positions in all scaffolds were turned into alanines without moving altering backbone coordinates with Rosetta3 fixbb protocol [26]. Each structure then was preprocessed as was described earlier.

Preprocessing and scaffold searching was carried out using the equipment of the shared research facilities of HPC computing resources at the Lomonosov Moscow State University ("Lomonosov-2" supercomputer) [27]. Preprocessing stage took 30 min 22 s on 64 cores with average preprocessing time for one scaffold of 3.66 s. Scaffold searching stage took 5 min 43 s on 64 cores with average search time for one scaffold of 0.69 s.

2.7 Rosetta Match Assessment

Structure of Porcine Pancreatic Trypsin (PDB ID 4DOQ) was used to assess the computational time of Rosetta Match application [10]. The theozyme included the catalytic triad Ser-His-Asp and a water molecule as a dummy substrate. The search was performed into the whole protein structure (221 residues). The -consolidate_matches flag was used to prevent massive and time-consuming output of nearly identical structures.

Rosetta Match was tested with -packing:ex1 and -packing:ex2 levels set to either default 1 or 3 for more precise rotamer sampling.

3 Results and Discussion

3.1 Backbone-Based Vectorization of Triads

We start by hypothesizing that for a scaffold searching task a theozyme for enzyme design may be in principle reduced to just the relative organization of backbones of crucial residues. Similar reduction was previously shown to be beneficial to the design of small-molecule binding sites [28]. Rationale for such reduction was given in the introduction section of the manuscript.

Another aspect that would benefit a scaffold searching problem is an ability to directly compare different backbone organizations by having a distance metric defined for such an object type. This notion requires a vectorization procedure as well. Trivial way to perform such vectorization is by expressing each triad as a vector of each of its atoms' coordinates. Once this is done, root mean square deviation of atomic positions (RMSD) is a natural measure of similarity between two such objects. However, such comparison requires an optimal superposition performed firsthand; what is more, such a description is redundant since it explicitly differentiates between translated and rotated copies of the same triad. It is possible to construct a more concise vectorization that

would be translation- and rotation-invariant and thus would not require preemptive superposition.

Backbone orientation of each residue may be represented as an oriented triangle with vertices N-CA-C (Fig. 1A). All measures of these triangles are fixed since the length of N-CA and CA-C bonds and the angle between them may be safely considered a constant for all protein structures. The vectorization task therefore is reduced to the problem of encoding the relative orientations of three such triangles. Taking rotational and translational invariance into account, only 12 degrees of freedom are left. For a pair of residues 6 values are sufficient to describe the relative orientations of their backbone: 5 angles defined in the Fig. 1B and the distance between any pair of their atoms. It is possible to construct an asymmetric triad vector by choosing a pivot residue and constructing two sets of 6 values each to explicitly encode the positions of two remaining residues. However, we decided to choose a different formalization in which each pair of residues forming a triad contributes 4 degrees of freedom, all expressed in angular or torsional form. Such vectorization is thus symmetric and uniform in data ranges and units which is useful for the calculation of distance between two such vectors without need for normalization (see Materials and Methods).

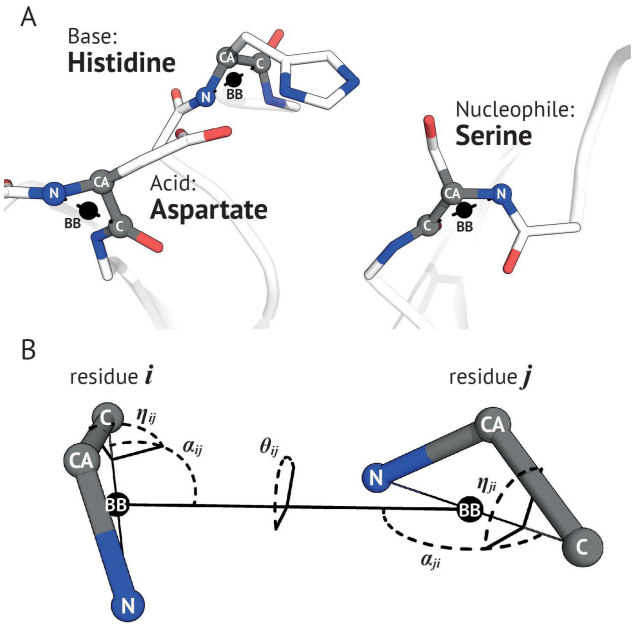

Fig. 1. Catalytic triad typical organization and vectorization. A. Architecture of the trypsin's catalytic triad. Backbone atom names are labeled and highlighted in gray. B. Vectorization introduced in current work.

3.2 Space of Architectures of Proteases' Catalytic Triads

We intended to investigate whether our simplistic approach is useful to describe the space of active site architectures. In this work we focused on catalytic triads of serine and cysteine endopeptidases. We found that clusterization based on our 12-dimensional vectorization produces highly informative insights, clearly distinguishing between different classes of proteolytic triads and non-catalytic triads (Fig. 2). The following clusters were formed: subtilisin-like architectures (Fig. 3A), trypsin-like architectures (Fig. 3B), papain-like architectures (Fig. 3C), caseinolytic protease-like (CLP-like), Backbone-based superposition to the cluster centers revealed that, indeed, backbone-only representation is sufficient to discriminate between various architectures more often described in terms of their sidechain relative orientations (Fig. 3). Our method was also sensitive enough to correctly assign a cluster label to the PDB entries harbouring substitutions in their active sites and ones covalently or noncovalently inhibited, even if sidechain geometry in these cases was distorted.

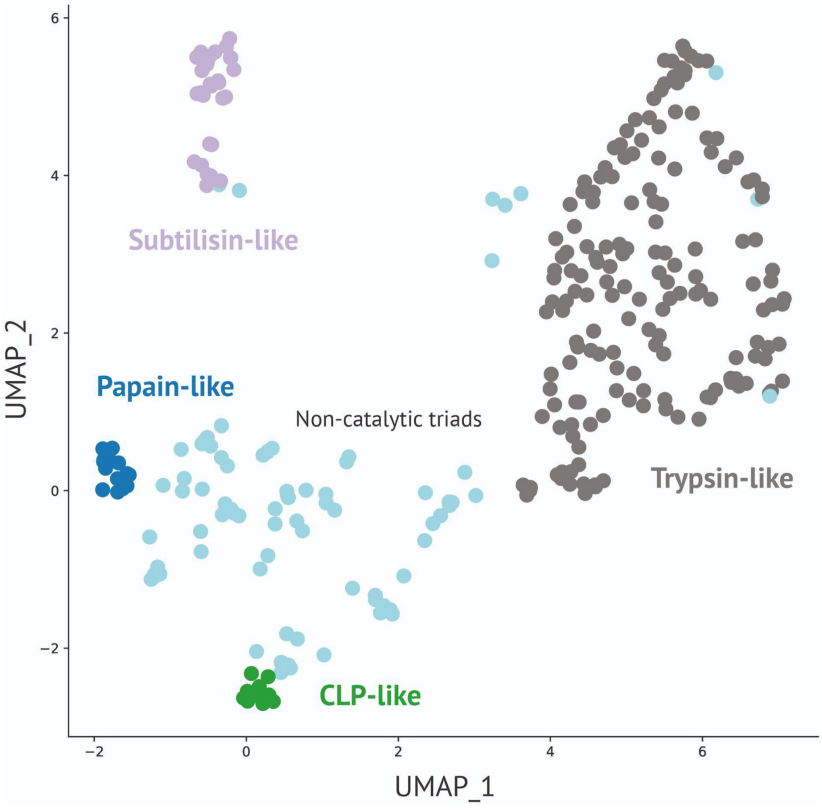

Fig. 2. Clusterization of catalytic triads architectures based on backbone vectorization.

Thus, a backbone-based approach was proven to not only be applicable to scaffold searching, but also to be a powerful tool to study the space of catalytic site architectures. Further generalization of the approach on different enzyme classes may produce new insight into the intricacies and evolutionary constraints of biocatalytic machineries.

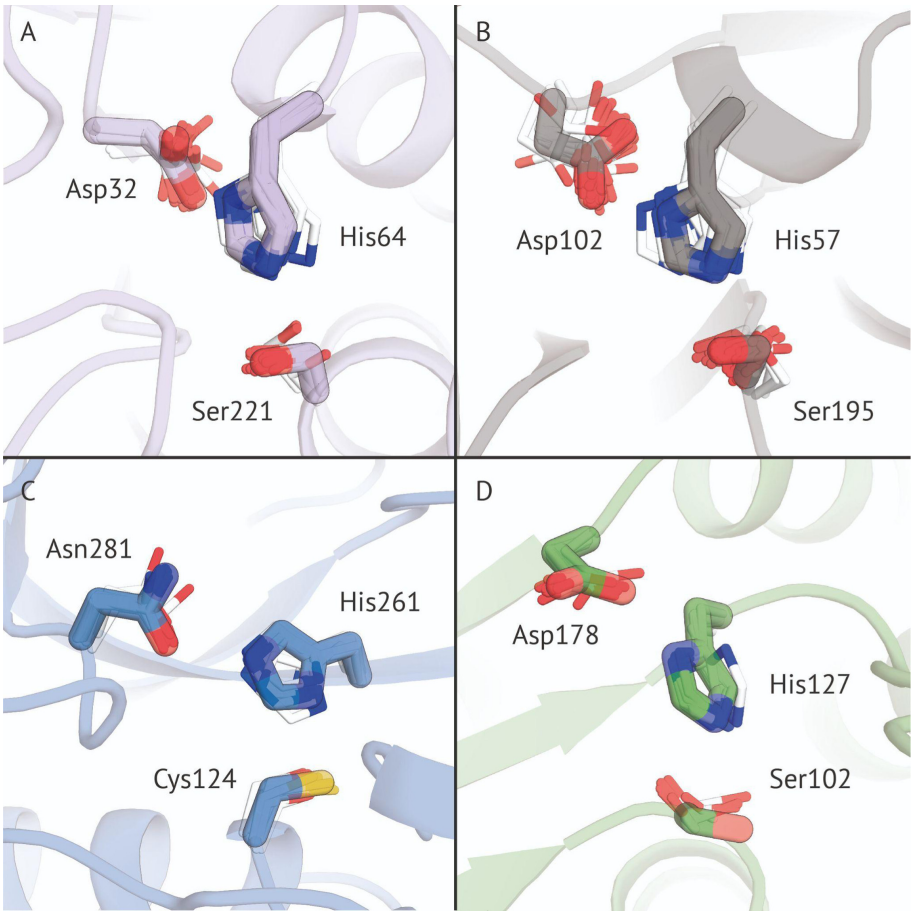

Fig. 3. Catalytic triad architecture of representatives of all clusters. A. Subtilisin-like cluster. Numbering is based on PDB ID 3BX1. B. Trypsin-like cluster. Numbering is based on PDB ID 1AVW. C. Papain-like cluster. Numbering is based on PDB ID 5Z5O. D. CLP-like cluster. Numbering is based on PDB ID 6NAH. Carbon coloration is in accordance with Fig. 2.

3.3 Scaffold Searching

We utilized our study of proteases to devise a distance threshold to be used to distinguish between adequate and inadequate placements, as well as some filters to reduce the number of scaffold triads to search through. We found that distributions of average distances to other cluster mates vary between triad architectures (Fig. 4), however always lying much lower than those of non-catalytic ones (minimal average distance of 224°). For the placement search for exact architecture type it is thus preferable to use a relevant threshold that we define as 90th percentile in the mean distances distribution within the cluster. However, due to the dramatic difference between catalytic and non-catalytic architectures, a milder threshold may be used, e.g. the maximum of the thresholds (in our case 47°, for trypsin-like triads).

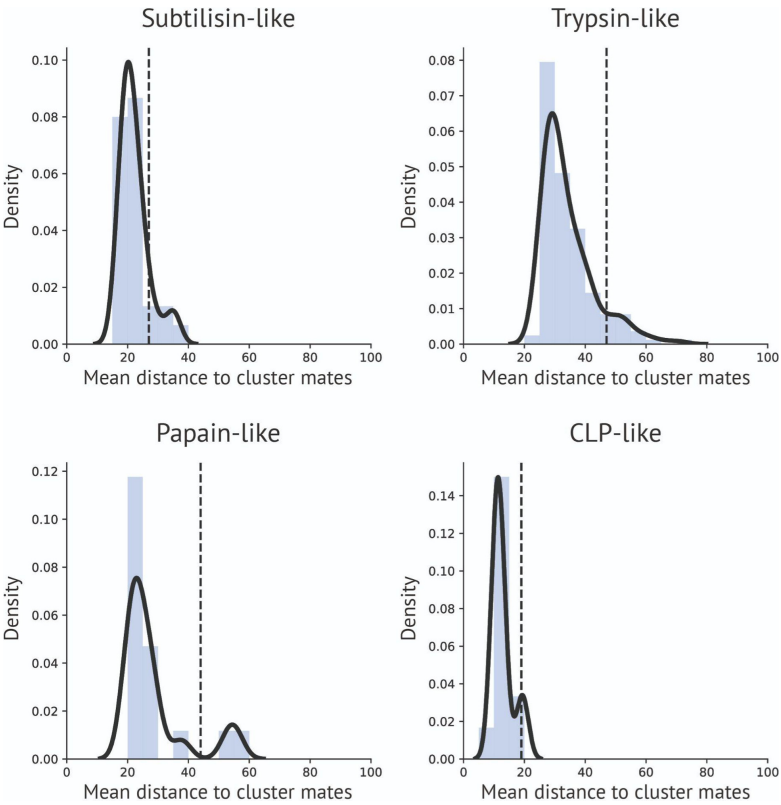

Fig. 4. Distributions of mean distances between each point in a cluster and every other point inside the same cluster. Upper-left: subtilisin-like triads, upper-right: trypsin-like triads, lower-left: papain-like triads, lower-right: CLP-like triads. Black dash represents the 90th percentile.

We further demonstrate our scaffold searching procedure on two examples: trypsin- and papain-templated search against a TEV protease scaffold, and trypsin-templated search against the whole CATH S40 non-redundant domains datased.

Prior to performing scaffold search we preprocessed each structure by converting it into a set of vectorized triads. To reduce the number of triads we applied several filters derived from the distributions studied for natural catalytic triads in proteases (Fig. 5).

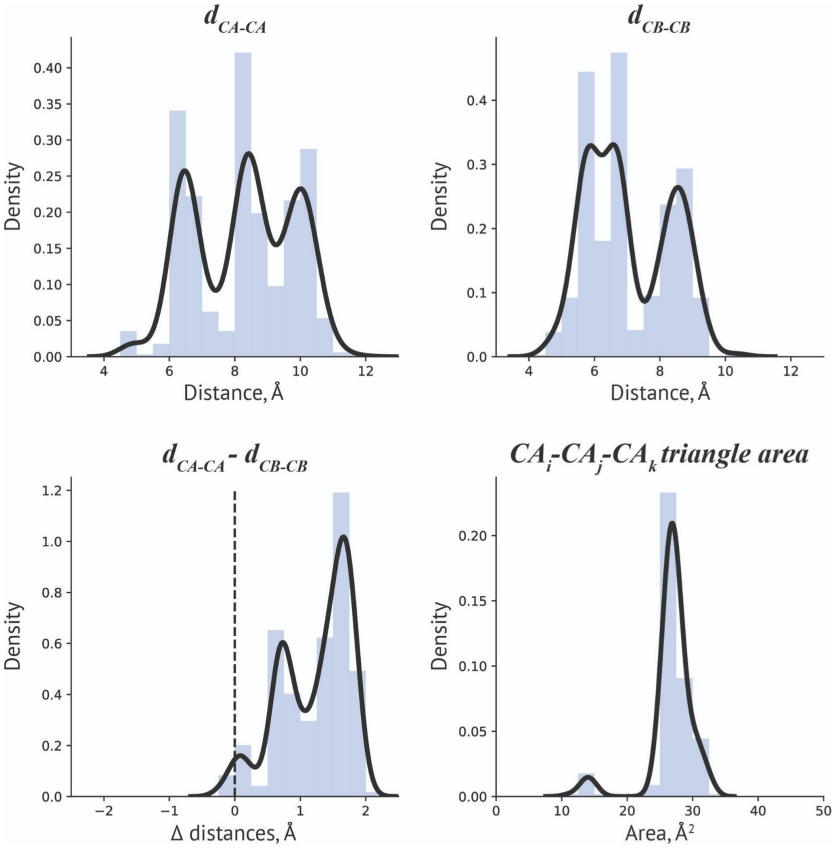

Fig. 5. Distributions of various auxiliary metrics useful for scaffold triads filtration prior to placement search. Upper-left: inter-CA-atomic distance, upper-right: inter-CB-atomic distance, lower-left: their difference, lower-right: area of the triangle built upon CA atoms of triad residues.

TEV protease is known to harbour a triad very much resembling that of trypsin despite being a cysteine protease [29]. On the other hand, it does not share much in common with papain-like architectures. Trypsin-templated search was able to easily identify the correct placement of TEV protease catalytic triad with the distance to it of 41.83° separated from all others by a significant margin (Table 1). On the other hand,

papain-templated search did not find any promising placements at all since all the best ones had a distance significantly higher than 47° from the reference vector (Table 2).

Table 1. Best 5 placements from trypsin-templated search against TEV protease scaffold.

Rank	Distance	Composition according to 1LVM chain A
1	*41.83°*	*Cys151, His46, Asp81 (catalytic triad)*
2	86.46°	Val112, Thr17, Ile14
3	87.50°	Leu190, Leu98, Phe37
4	96.64°	Phe37, Leu189, Phe186
5	97.71°	Leu190, Phe94, Phe37

Both these searches were performed under 2 s on a single core. We decided to compare the computational effectiveness of our approach with those of Rosetta Match on a trivial case of trypsin-templated search against trypsin scaffold. Naturally, both methods succeeded in correctly identifying an ideal placement. However, it took Rosetta Match 42 s to perform the task with a standard level of rotamer sampling and 1 m 10 s with sampling extended to 3σ. Extending the number of samples per constraint skyrockets the computational time beyond 1 day. Our backbone-vectorization based approach took just 1.29 s. This comparison clearly shows the strength and practical applicability of our approach.

As an example of a near real-world application we performed a search against the whole CATH S40 non-redundant domains datased. It took on average 0.69 s to scan through all the possible placements inside a scaffold. In total, 16 placements with distance below 47° were found (Table 3).

Table 2. Best 5 placements from papain-templated search against TEV protease scaffold.

Rank	Distance	Composition according to 1LVM chain A
1	82.65°	Gly152, Tyr33, Ser15
2	94.64°	Ile144, Cys110, Val125
3	97.70°	Gly152, Try33, Ile18
4	101.90°	Ile84, Leu189, Ile42
5	111.13°	Gly152, Tyr33, Thr113

Unsurprisingly, the top of the table is occupied by other proteases. Starting from the 8th hit, 1AUK with a distance of 39.72°, is a transition towards non-proteolytic folds. Whether they can in fact be successfully engineered into proteases utilizing the recommendations from the scaffold search is a matter of further study. If so, the recommended threshold at 90th percentile is indeed a reasonable assumption. We note however that a protein designer may want to search for looser matches if one has means of computational backbone reengineering at a disposal. The used dataset is also only

Table 3. Hits (distance <47°) from trypsin-templated search against CATH S40 domains dataset.

Rank	Distance	PDB ID	Positions	Comment
1	16.96°	3OTP	Ala210, His105, Asp135	Protease, S > A mutant
2	18.73°	1WXR	Ser207, His73, Asp101	Protease
3	25.98°	4BXS	Ser362, His211, Asp265	Protease
4	29.60°	2F83	Ser557, His413, Asp462	Protease
5	36.18°	3H09	Ser288, His100, Asp164	Protease
6	37.94°	3SZE	Ala263, His127, Asp156	Protease, S > A mutant
7	38.87°	4B6E	Ser139, His57, Asp81	Protease
8	*39.72°*	*1AUK*	*Gly292, Glu285, Asp30*	*Not a protease*
9	39.81°	4M9F	Ser1135, His1051, Asp1075	Protease
10	*40.52°*	*4AKF*	*Ala72, Gly289, Ser268*	*Not a protease*
11	42.67°	2WV9	Ser135, His51, Asp75	Protease
12	*42.68°*	*1WKB*	*Thr319, Pro226, Trp263*	*Not a protease*
13	*42.79°*	*1NFV*	*Leu37, Leu45, Ile157*	*Not a protease*
14	*44.00°*	*3H75*	*Ala285, Phe273, Pro267*	*Not a protease*
15	*44.67°*	*3TGH*	*Gln115, Thr101, Trp144*	*Not a protease*
16	*46.42°*	*3QZ0*	*Val56, Lys78, Arg95*	*Not a protease*

partially reflecting real-world enzyme design studies since more specific, potentially *de novo* modeled scaffolds may be of better use to scan for. Concreticising use-cases as well as fine-tuning the filters and adding new ones is certainly needed in order to turn the presented approach into a tool or a web-service that can be accessed by a global community.

4 Conclusions

In the presented study a simplistic approach to the scaffold search problem of *de novo* enzyme design is proposed and validated. We show that by reducing the problem to the level of relative backbone orientations we can achieve a dramatic speed-up compared to existing approaches while producing meaningful results. Our solution makes it possible to routinely scan the whole structural proteome for promising placements of catalytic architectures on a working station or a small cluster. What is more, proposed vectorization allows to uncover hidden patterns in the organization of enzymes that may lead to new fundamental discoveries in the field of structural enzymology.

Acknowledgements. This work was supported by Russian Science Foundation Grant 21-74-20113. This research has been conducted in frame of the Interdisciplinary Scientific and Educational School of Moscow State University “Molecular Technologies of the Living Systems and Synthetic Biology”.

References

1. López-Otín, C., Bond, J.S.: Proteases: multifunctional enzymes in life and disease. J. Biol. Chem. **283**, 30433–30437 (2008)
2. Di Cera, E.: Serine proteases. IUBMB Life **61**, 510–515 (2009)
3. Brömme, D.: Papain-like cysteine proteases. Curr. Protoc. Protein Sci. **Chapter 21**, Unit 21.2 (2001)
4. Leis, J.P., Cameron, C.E.: Engineering proteases with altered specificity. Curr. Opin. Biotechnol. **5**, 403–408 (1994)
5. Lau, Y.-T.K., et al.: Discovery and engineering of enhanced SUMO protease enzymes. J. Biol. Chem. **293**, 13224–13233 (2018)
6. Chowdhury, R., Maranas, C.D.: From directed evolution to computational enzyme engineering—a review. AIChE J. **66** (2020)
7. Vaissier Welborn, V., Head-Gordon, T.: Computational design of synthetic enzymes. Chem. Rev. **119**, 6613–6630 (2019)
8. Pultz, I.S., et al.: Gluten degradation, pharmacokinetics, safety, and tolerability of TAK-062, an engineered enzyme to treat celiac disease. Gastroenterology (2021)
9. Mokrushina, Y.A., et al.: Multiscale computation delivers organophosphorus reactivity and stereoselectivity to immunoglobulin scavengers. Proc. Natl. Acad. Sci. U. S. A. **117**, 22841–22848 (2020)
10. Richter, F., Leaver-Fay, A., Khare, S.D., Bjelic, S., Baker, D.: De novo enzyme design using Rosetta3. PLoS ONE **6**, e19230 (2011)
11. Linder, M.: Computational enzyme design: advances, hurdles and possible ways forward. Comput. Struct. Biotechnol. J. **2**, e201209009 (2012)
12. Siegel, J.B., et al.: Computational design of an enzyme catalyst for a stereoselective bimolecular Diels-Alder reaction. Science **329**, 309–313 (2010)
13. Jiang, L., et al.: De novo computational design of retro-aldol enzymes. Science **319**, 1387–1391 (2008)
14. Zanghellini, A.: de novo computational enzyme design. Curr. Opin. Biotechnol. **29**, 132–138 (2014)
15. Freiberger, M.I., Guzovsky, A.B., Wolynes, P.G., Parra, R.G., Ferreiro, D.U.: Local frustration around enzyme active sites. Proc. Natl. Acad. Sci. U. S. A. **116**, 4037–4043 (2019)
16. Ferreiro, D.U., Komives, E.A., Wolynes, P.G.: Frustration in biomolecules. Q. Rev. Biophys. **47**, 285–363 (2014)
17. Whiting, A.K., Peticolas, W.L.: Details of the acyl-enzyme intermediate and the oxyanion hole in serine protease catalysis. Biochemistry **33**, 552–561 (1994)
18. Ménard, R., Storer, A.C.: Oxyanion hole interactions in serine and cysteine proteases. Biol. Chem. Hoppe Seyler. **373**, 393–400 (1992)
19. Wang, G., Dunbrack, R.L., Jr.: PISCES: a protein sequence culling server. Bioinformatics **19**, 1589–1591 (2003)
20. Bakan, A., et al.: Evol and ProDy for bridging protein sequence evolution and structural dynamics. Bioinformatics **30**, 2681–2683 (2014)
21. Garreta, R., Moncecchi, G.: Learning Scikit-Learn: Machine Learning in Python. Packt Publishing Ltd. (2013)
22. McInnes, L., Healy, J., Saul, N., Großberger, L.: UMAP: uniform manifold approximation and projection. J. Open Sour. Softw. **3**(29), 861 (2018)
23. The PyMOL Molecular Graphics System, Version 2.0 Schrödinger, LLC

24. Hagberg, A., Schult, D., Swart, P.: Exploring network structure, dynamics, and function using Networkx. In: Varoquaux, G., Vaught, T., Millman, J. (eds.). Proceedings of the 7th Python in Science Conference 2008, Pasadena, CA USA, pp. 11–15 (2008)
25. Sillitoe, I., et al.: CATH: comprehensive structural and functional annotations for genome sequences. Nucleic Acids Res. **43**, D376–D381 (2015)
26. Leaver-Fay, A., et al.: ROSETTA3: an object-oriented software suite for the simulation and design of macromolecules. Methods Enzymol. **487**, 545–574 (2011)
27. Supercomputer lomonosov-2: large scale, deep monitoring and fine analytics for the user community. Supercomput. Front. Innov. **6** (2019)
28. Polizzi, N.F., DeGrado, W.F.: A defined structural unit enables de novo design of small-molecule-binding proteins. Science **369**, 1227–1233 (2020)
29. Phan, J., et al.: Structural basis for the substrate specificity of tobacco etch virus protease. J. Biol. Chem. **277**, 50564–50572 (2002)