Masaryk University

Faculty of Science

Computational Tools for Analysis and Design of Proteins

Habilitation Thesis

David Bednář

Brno 2022

Acknowledgments:

I would like to thank my past supervisors for their valuable advice and for forming my scientific mind. My special thanks belong to Jiří Damborský for the possibility of working in Loschmidt Laboratories and collaborating on so many exciting projects.

I want to thank also all my colleagues and all co-authors for our fruitful work together. Without their vast contribution, it would not be possible to finish any of the projects.

Last but not least, I would like to thank my parents, my wife Petra, and son Honzík for their love and unlimited support.

Content

COMMENTARY

This Habilitation Thesis is a compilation of selected scientific publications with my contribution as author or co-author. The results were obtained mainly at Masaryk University in Brno, Czech Republic and articles were published between 2017 and 2022. The research aimed at the development of methods and computational tools for the engineering and analysis of proteins. The focus was both on developing novel algorithms and on bringing different structure- and sequence-based analyses to the broader scientific community or even inexperienced users. Therefore, the emphasis was put on understandable workflows coupled with easy-to-use graphical user interfaces.

This Thesis is divided into two parts. Part I contains the introduction to individual scientific problems, current state-of-the-art, and contributions to the field by individual tools for protein analysis and engineering. Part II consists of 10 publications (listed below) where all selected tools are thoroughly described. The topics of these studies cover searching for novel enzymes, protein solubility, protein stability, and analysis of access pathways.

Publications covered by this Habilitation Thesis:

1. Hon, J., Borko, S., Stourac, J., Prokop, Z., Zendulka, J., **Bednar, D.**, Martinek, T., Damborsky, J., 2020: EnzymeMiner: Automated Mining of Soluble Enzymes with Diverse Structures, Catalytic Properties and Stabilities. ***Nucleic Acids Research*** 48: W104–W109.
2. Hon, J., Marusiak, M., Martinek, T., Kunka, A., Zendulka, J., **Bednar, D.**, Damborsky, J., 2021: SoluProt: Prediction of Soluble Protein Expression in Escherichia coli. ***Bioinformatics*** 37: 23-28.
3. Musil, M., Stourac, J., Bendl, J., Brezovský, J., Prokop, Z., Zendulka, J., Martinek, T., **Bednar, D.**, Damborsky, J., 2017: FireProt: Web Server for Automated Design of Thermostable Proteins. ***Nucleic Acids Research*** 45 (W1): W393-W399.
4. Musil, M., Khan, R. T., Beier, A., Stourac, J., Konegger, H., Damborsky, J., **Bednar, D.**, 2021: FireProtASR: A Web Server for Fully Automated Ancestral Sequence Reconstruction. ***Briefings in Bioinformatics*** 22: 1-11.
5. Stourac, J., Dubrava, J., Musil, M., Horackova, J., Damborsky, J., Mazurenko, S., **Bednar, D.**, 2020: FireProtDB: Database of Manually Curated Protein Stability Data. ***Nucleic Acids Research*** 49: D319-D324.
6. Jurcik, A., **Bednar, D.**, Byska, J., Marques, S. M., Furmanova, K., Daniel, L., Kokkonen, P., Brezovsky, J., Strnad, O., Stourac, J., Pavelka, A., Manak, M., Damborsky, J., Kozlikova, B., 2018: CAVER Analyst 2.0: Analysis and Visualization of Channels and Tunnels in Protein Structures and Molecular Dynamics Trajectories. ***Bioinformatics*** 34: 3586-3588.
7. Vavra, O., Filipovic, J., Plhak, J., **Bednar, D.**, Marques, S.M., Brezovsky, J., Stourac, J., Matyska, L., Damborsky, J., 2019: CaverDock: A Molecular Docking-Based Tool to Analyse Ligand Transport through Protein Tunnels and Channels. ***Bioinformatics*** 35: 4986-4993.
8. Stourac, J., Vavra, O., Kokkonen, P., Filipovic, J., Pinto, G., Brezovsky, J., Damborsky, J., **Bednar, D.**, 2019: Caver Web 1.0: Identification of Tunnels and Channels in Proteins and Analysis of Ligand Transport. ***Nucleic Acids Research*** W1: W414–W422.
9. Sumbalova, L., Stourac, J., Martinek, T., **Bednar, D.**, Damborsky, J., 2018: HotSpot Wizard 3.0: Web Server for Automated Design of Mutations and Smart Libraries Based on Sequence Input Information. ***Nucleic Acids Research*** 46: W356-W362.

PART I

INTRODUCTION

1 Searching for Novel Enzymes

1.1 Introduction

During the billion years of evolution, nature has developed an enormous diversity of biomolecules. Thanks to the next-generation sequencing technologies[1,2], sequences of these biomolecules became available and provided an excellent source of novel protein sequences. Many genome and metagenome sequencing projects are running at an incredible pace resulting in significant growth of sequence databases[3,4]. The number of sequences is doubling every 2.5 years[5]. On the other hand, experimental characterization of individual proteins via classical biochemical techniques is very costly and time-demanding. This huge contrast between sequenced and characterized proteins is nicely visible on the UniProt database (https://www.uniprot.org). The automatically annotated part of UniProt, TrEMBL database, contains about 220 million entries, whereas there are only about 570 thousand sequences in manually reviewed Swiss-Prot (data from 07/2021). Recently, several successful cases where high-through screening techniques, like robotic or microfluidic platforms, were developed to identify enzymes with convenient properties[6–8]. Despite this significant progress in high-throughput techniques, these methods are still rather scarce, and their development for a particular application can be very time-demanding. Therefore, new approaches with sufficient capacity to screen or prioritize attractive enzymes are still highly valued. *In silico* approaches provide an appropriate solution for screening large databases incomparably faster than any experimental technique.

1.2 State-of-the-art

There are more than 220 million uncharacterized sequences in the UniProt database[3], many of them with enzymatic functions, which can be of great value in many biotechnological or medical applications. Genome Online Database (GOLD) database[9] currently contains about 150 thousand ongoing sequencing projects, so the number will grow steadily in the future. The most significant disadvantage of these databases is that

most sequences include only insufficient automatically derived annotations that lack information about their biological function. Therefore many tools and databases have to be used to filter out biologically relevant information.

Current approaches in novel enzyme identification usually start with metadata search or sequence similarity search of a known characterized enzyme against UniProt or GeneBank databases[10]. Metadata, like protein name, functional annotations, source organism, etc., are mostly automatically annotated, and their accuracy is relatively low. Therefore, further analyses and filtering are necessary to obtain good-quality data. On the other hand, sequence similarity search is based only on the evolution similarity of homologous sequences and not on the accuracy and availability of individual user-defined metadata. Therefore, the sequence-based search leads to a much higher specificity. First algorithms were based on an optimal deterministic search like Needleman and Wunch[11], but with the growing size of the databases, much faster heuristic algorithms are utilized. BLAST algorithm[12] is usually applied for pairwise sequence alignment as the golden standard method. BLAST is based on finding minimal alignment defined by word size, which is then extended into a full-length alignment. The identity of sequences is scored by the number of substitutions and gaps specified by scoring matrices[13]. Even though it is one of the oldest heuristic algorithms, it is still broadly used for sequence similarity search. Suppose more search sensitivity is needed, algorithms based on position-specific sequence matrices, like PSI-BLAST[14], or profile-based methods, like HMMER[15], are able to find even more distant homologs than basic BLAST search.

Sequence similarity search usually provides an overwhelming amount of hits and must be accompanied by analyses using computational tools and biological databases to obtain more information on individual sequences. These analyses can provide valuable annotations on sequence properties[16], structure motifs, enzyme function[17], classifications[18], and localization[19], or information on source organisms and their natural environment[20]. These annotations can help with further filtering and prioritization of biotechnologically attractive enzymes.

1.3 Contribution to the field

Sequence similarity searches provide hits with relatively low specificity. Therefore, as mentioned above, many tools and databases have to be utilized to decrease the number of identified putative enzymes to levels reasonable for experimental validation. The more the *in silico* analyses can be applied, the better the chance to find stable, functional, and expressible enzymes. Unfortunately, many tools need to be installed, run, and adequately analyzed, which often require non-trivial knowledge in bioinformatics. Therefore, we designed a computational pipeline capable of searching, filtering, and annotating novel enzymes available in sequence databases with a user-defined function. The EnzymeMiner web server with a user-friendly graphical user interface was developed to bring this analysis to the broader scientific community. The only input is the sequence of an enzyme with the required function and the list of essential, i.e. catalytic or ligand-binding residues. We believe the tool can allow experimental biologists to find novel enzymes with interesting properties without the need for running many different analyses (Figure 1).

The current version of EnzymeMiner provides only sequence-based information about the proteins in the nr database of NCBI. Recent development in the sequencing of metagenomes and the availability of their results in the MGnify database[4]provides a new source of potentially interesting enzymes which will more than double the current amount of available sequences. Moreover, the recent publication of the structure prediction tool, AlphaFold 2[21], opens the doors for structure-based analyses, providing an entirely new level of annotations. Both structure analysis and metagenomes search will be utilized during the development of the latest version of EnzymeMiner. The current version of the EnzymeMiner web server is freely available at https://loschmidt.chemi.muni.cz/enzymeminer/ and is thoroughly described in PART II.

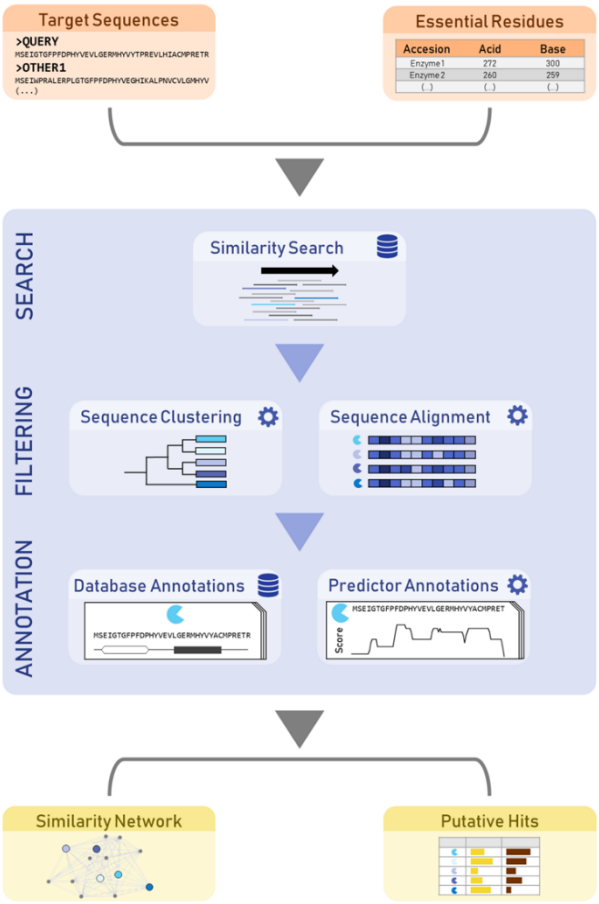

Figure 1. Illustration of the EnzymeMiner workflow. The workflow consists of input data (orange), search, filtering and annotation of homologous sequences (blue), and visualization of results (yellow). Adopted from Vasina et al.[22]

2 Protein Solubility Prediction

2.1 Introduction

Solubility is an essential attribute of every protein. Insufficient solubility is one of the leading causes of failure in protein production and is critical for any protein-related work. Therefore, the prediction of solubility by computational tools is of great interest in both basic science and industry. By definition, protein solubility is a thermodynamic parameter defined as the protein concentration in a saturated solution in equilibrium with a solid phase, either crystalline or amorphous, under a given set of conditions[23]. However, experimental measurement of this parameter on a large scale is rather difficult. Moreover, it does not capture all the problems preventing recombinant protein production in a functional state in a high concentration. Therefore, *expressibility* or *soluble expression* is a more precise therm where the final *soluble* protein must be overexpressed in the soluble, well-folded form[24,25].

Moreover, many other problems hamper protein solubility prediction *in silico* primarily due to the inconsistency between available experimental data. Many conditions influence the proper protein solubility, like protein concentration in the cell, presence of chaperons, type of the expression system, physical and chemical factors acting during the expression, or cell protective mechanisms acting against toxicity of some proteins[26]. Therefore, many solubility predicting tools have to either focus on prediction in very specific conditions or neglect the inconsistencies between the expression conditions. Thus, the resulting tools either work on only a very limited number of cases or have overly low accuracy. Because the prediction of protein solubility is a very complex problem and we are not able to define all the crucial parameters by physico-chemical means, machine learning is usually utilized to build functional models.

2.2 State-of-the-art

The accuracy of the sequence-based prediction of machine learning-based methods highly depends on the size and quality of data available for both training and testing. The largest public source of experimental data is the TargetTrack database[27] provided by Protein Structure Initiative projects. TargetTrack contains experimental data from crystallization trials of about 900,000 proteins. Even though the database is not primarily focused on protein solubility, the information on whether the protein can be obtained in purified form has a great value for designing a large, well-balanced dataset. On the other hand, low quality of annotations, lack of consistent information on expressibility failures, and plain text format of individual entries make the database unusable without significant manual curation and filtering. Several other databases can be used in solving particular problems, but none of them has the potential to be used as a large and diverse source of training data: i) NESG[28] containing high-quality data on protein solubility of almost 10,000 proteins expressed in *E. coli,* is a good candidate for testing dataset, ii) HGPD[29] contains over 9,000 measurements of human proteins, iii) AMYPdb[30] with data about 12,000 amyloid precursor proteins, iv) eSOL[31] containing over 4,000 entries from cell-free expression systems, or v) RCSB PDB[32,33] containing about 190,000 proteins structures but, by the definition of the database, only of soluble proteins.

Solubility depends on many factors, including all the extrinsic conditions (concentration, pH, temperature, expression system, chaperons) but also intrinsic properties defined by amino acid composition. This makes the prediction models too complex to be built rationally. Therefore, the vast majority of available tools is based on statistical analysis or machine learning. Solubility predictors depend on the features which can be extracted or predicted at the protein sequence level. The most common features utilized by many current tools belong to these categories: i) amino acid content depending on frequencies of individual residues, dimers, or trimers, ii) physical-chemical properties of amino acids (a charge, hydrophobicity, polarity, size, etc.) mainly obtained from AAindex database[34], iii) features predicted by other tools (secondary structure, solvent accessibility,

or disordered and transmembrane regions), and iv) sequence identity based on pairwise alignment towards known soluble or insoluble proteins.

Currently, there are many tools available utilizing the discussed sequence-based features to predict protein solubility. These tools differ in feature prioritization and statistical or machine learning models used for the prediction. Based on the model, tools can be divided into i) discriminant analysis (revised Wilkinson-Harrison model), ii) linear regression (CamSol, Protein-Sol), iii) logistic regression (RPSP, PROSSO II, SWI), iv) Support Vector Machine (SOLpro, ccSOL, ESPRESSO), and v) neural network (DeepSol, SKADE). Interestingly, not necessarily the more sophisticated models provide better results. When tested on an independent testing dataset, many of the tools do not provide the accuracies presented by the authors (Figure 2). Therefore, it is challenging to select the best predictors based on the original papers due to overtraining or inadequate testing.

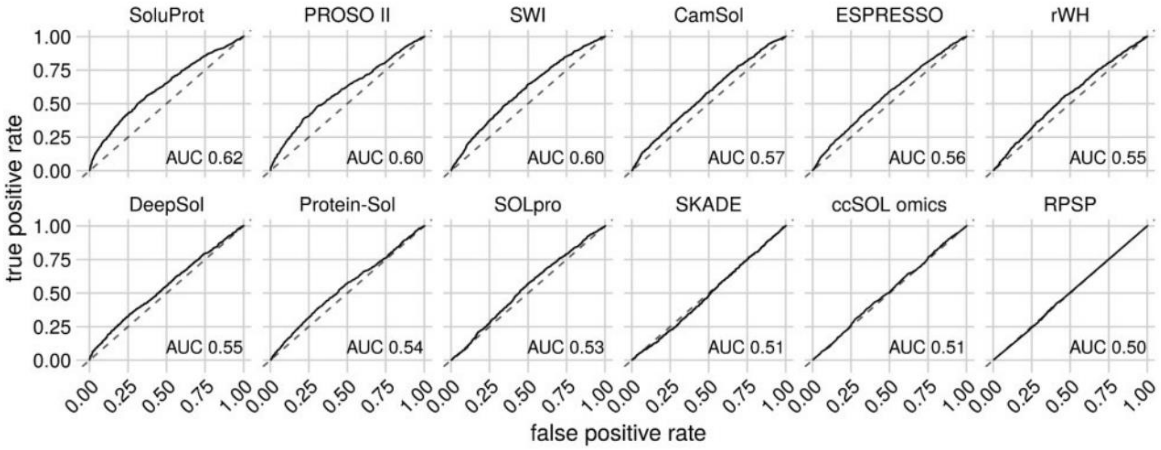

Figure 2. Receiver operating curves of sequence-based solubility predictors tested on SoluProt test set of 3100 sequences. Tools are ordered by the area under the curve (AUC). Adopted from[35].

An even more pronounced problem represents the prediction of the change in protein solubility caused by a mutation. The ability to predict the effect of a mutation would be of great interest in industry because sufficient protein production is often limiting for many applications. Unfortunately, there is only a limited amount of data as no database gathering the information about changes in solubility upon mutation is available. The only

datasets that can be obtained are presented as part of the developed tools, but they contain only tens or lower hundreds of entries[36–38]. Individual tools are then based only on the sequence analysis, employing machine learning (PON-sol[36] or requires 3D structure either as optional (CamSol[37] and SODA[39]) or mandatory input (SolubiS[40,41], AGGRESCAN3D[42], and OptSolMut[38]). These methods then combine profile-based solubility prediction with structure corrections, stability evaluation, or introduce a scoring function for solubility calculation. Given the limited data available, training of machine-learning-based models or parametrization of the structure-based features may be biased towards the small number of proteins represented in the dataset. Standardizing the mutational data and their FAIR storing would help the developers make their tools more generally applicable with improved accuracy.

2.3 Contribution to the field

Good solubility is one of the basic parameters for any protein used in both basic science and industry. Current predictors often do not possess the outstanding accuracy presented by their developers when tested on an independent dataset[35]. Therefore, we designed a new tool called SoluProt for sequence-based solubility prediction of proteins produced in the most common host organism, *E. coli*. SoluProt is a machine-learning-based tool employing random forest regression. It was trained on a dataset of more than 11,000 data points obtained by filtration and manual curation of the TargetTrack database. SoluProt and 11 other predictors were tested on a dataset based on the NESG database containing 3,100 data points. Even though the prediction power is still relatively low, SoluProt outperformed all the other tools by accuracy, Matthew's correlation coefficient, or area under the ROC curve. Similar values were observed only for PROSSO II, but in this case, the testing dataset significantly overlapped (95 %) with the training set of PROSSO. Therefore, the final results could be biased towards better performance.

SoluProt was provided to the community as a web server freely available at https://loschmidt.chemi.muni.cz/soluprot/ and is thoroughly described in PART II. Moreover, solubility was also observed as the most problematic property when mining

novel enzymes[43,44]. Therefore, SoluProt was introduced into the pipeline and the selection table of EnzymeMiner as another criterium for the selection of functional enzymes.

To tackle the problem of predicting solubility change upon mutation, high-quality data are necessary both for training and testing novel methods. Currently, there is no database of mutational effects on solubility available. A few datasets constructed by tool developers are rather scarce, contain errors, and do not cover data from novel high-throughput assays[45,46]. Therefore, we developed a database called SoluProtMut^{DB}, which would help with gathering available data, their curation, and standardization. Currently, it contains more than 32,000 mutants of about 100 proteins. Together with our data from high-throughput microfluidic platforms that are presently being produced, it can create a sufficient dataset for training and testing novel tools. SoluProtMut^{DB} is freely available at https://loschmidt.chemi.muni.cz/soluprotmutdb.

3 Prediction of Protein Stability

3.1 Introduction

Stability is another fundamental property to be considered in both basic science and industrial applications. It can be driven either by thermodynamics (the most stable conformation) or kinetics (the most accessible conformation) and depends on many intrinsic and extrinsic factors. Most natural proteins are thermodynamically only marginally stable up to the level that is necessary for their function[47], and in non-standard conditions, they are losing their proper structure rapidly[48]. On the other hand, high stability is correlated with high thermostability[49], resistance to pH or denaturing chemicals[50], prolonged half-life[51], serum survival time[52], good expression yields[53], and with resistance towards effect of mutations[54]. Therefore, stability improvements are important in biotechnology, medicine, the food industry, and biocatalysis and provide stable templates for protein engineering.

Experimental methods like directed evolution or saturation mutagenesis are capable of providing highly stable designs[55–57]. The drawback is the necessity of screening or selection techniques that make them time-consuming and cost-ineffective. Moreover, the setup of these methods is sometimes too difficult to be generally applicable to every protein. Therefore, computational methods are increasingly applied to pre-select positions[58] or identify particular mutations[59] to accelerate the screening process and reduce the experimental effort to the minimum.

3.2 State-of-the-art

With a significant increase in computational power in the past few decades, there was also a boom in the development of tools for predicting stability change upon mutation. These methods can be divided into three categories: i) prediction of free energy change, ii) analysis of evolutionary information, and iii) machine learning.

Force-field-based methods evaluating the change in folding free energy are probably the most common. Universal force fields like in Rosetta[60], FoldX[61,62], or Eris[63] are robust and have very good accuracy. On the other hand, the necessity of having a high-quality 3D structure as an input and the high demand for computational resources and time make them not applicable generally to every protein.

Phylogeny-based methods like consensus design (CD) or ancestral sequence reconstruction (ASR) do not have these limitations because they use only sequence input of target and homologous proteins. CD is available in several tools which are easy to use (3DM, VectorNTI, and HotSpot Wizard). The drawback of CD methods is that many of the consensual mutations preserve also other properties and are not stabilizing[64]. Therefore, these methods rather create a pool of mutations that must be further filtered or tested experimentally. ASR was proven to construct very robust proteins with good stability, activity, selectivity, and expressibility[65–68], but their success depends strongly on high-quality multiple sequence alignment (MSA) and phylogenetic tree. Even though there are tools that provide ASR design (RAxML, FastML, HandAlign), all of them leave the essential steps of selection of homologous sequences and MSA construction to the user.

The advantages of methods based on machine learning are the speed and the ability to recognize unknown dependencies that are not considered in force fields. Any characteristic that can describe the data can be used as a feature and potentially improve the predicting power. Unfortunately, these methods suffer from insufficient quality, size, and diversity of data that can be used for training and testing. Most of the machine learning tools were based on the ProTherm database[69] because, for a long time, it was the only available source of stability data. It was updated in 2020 and contains about 31,500 entries. Unfortunately, about 40 % are data from wild types and many inaccuracies in annotations, inconsistencies in values, and errors in signs were reported[70,71]. Software developers usually need to filter and manually curate ProTherm data so it can be applied for training or testing purposes[70,72]. Moreover, the prediction capabilities of individual tools are hard to compare based on the original papers. Datasets used for training and testing are small

or lack protein diversity. Therefore, many machine learning tools are biased towards proteins in their training datasets and fail in general accuracy[60,73,74]. A thorough description of stability prediction tools and datasets we discuss in the review by Musil et al.[75]

3.3 Contribution to the field

Currently, the most successful methods for protein stability design are hybrid approaches combining more above-mentioned strategies. They usually rely on force field calculations combined with either short molecular dynamics simulations and cysteine bridge design (FRESCO[76]) or with evolution-based methods (FireProt[77], PROSS[53]). Our method FireProt was the first that focused on the direct design of multiple-point mutants. It is divided into two separate branches, one using Rosetta and FoldX stability evaluation of all possible mutations and the second applying back-to-consensus analysis. Both strategies are accompanied by statistical, geometry, and energy filters. The effectivity of FireProt was shown on the stabilization of haloalkane dehalogenase DhaA and dehydrochlorinase LinA, both stabilized by more than 20 °C[77]. Later, the FireProt method was transformed into an automated, user-friendly web server enabling protein stability design using both branches of the original method. The tool was proven to be effective by the community of users who successfully stabilized their proteins[78–81]. FireProt is available at https://loschmidt.chemi.muni.cz/fireprotweb/ and is thoroughly described in PART II.

Phylogeny-based methods provide a great alternative when the structure of the target protein is not available. Consensus design is not very specific and has been implemented already in several tools, but ancestral sequence reconstruction was still far from being used automatically. Therefore, we developed FireProt^{ASR}, a web server that fully automizes the whole ASR workflow under one interface. The tool uses the protein sequence as the only input. It provides all ASR steps, like gathering homologous sequences, clustering and filtering, construction of MSA and phylogenetic tree, rooting of the tree, reconstruction of ancestral sequences, and taking care of the ancestral gaps correction. The tool was tested on the enzyme DhaA from the haloalkane dehalogenase family. Five out of six designs tested experimentally were folded correctly, stabilized by 20-26 °C, and retained

or improved activity and yields. FireProt^{ASR} is available at https://loschmidt.chemi.muni.cz/fireprotasr/ and is thoroughly described in PART II.

By the time we released FireProt^{DB}, a database of protein stability mutants, the available source of stability data was the outdated database ProTherm. ProTherm was last updated in 2013, and many inconsistencies were reported, including redundancy, missing or wrong values, or opposite signs of ΔΔG. The second database available, ProtaBank from 2018, introduced new data but did not solve the problems ProTherm contained. FireProt^{DB} contains experimental data of 16,000 mutants of more than 300 protein structures. The database comprises data from ProTherm, stability data from ProtaBank, data from a recent literature search, and experimental data from our group. Moreover, more than 10,000 mutations were verified in the original publications to reduce the errors in signs, values, and other inconsistencies. FireProt^{DB} is available at https://loschmidt.chemi.muni.cz/fireprotdb/ and is thoroughly described in PART II.

4 Analysis of Ligand Pathways

4.1 Introduction

It is estimated that more than 60 % of all enzymes have their active sites buried inside the protein core, forming an optimal microenvironment for particular functions[82]. These buried active sites are connected with the surrounding bulk solvent through molecular pathways, usually referred to as tunnels. Tunnels exist in all enzyme classes and enable the transport of substrates, cofactors, solvents, and products to and from the active site. Therefore, they play an essential role in the enzyme's catalytic cycle because the tunnel-lining residues can significantly influence the rates of substrate binding, product release, or substrate inhibition[83–85]. Moreover, tunnel opening, dynamics, and amino acid composition can help to distinguish between ligands of different sizes, flexibility, and physico-chemical properties and provide an additional level of enzyme selectivity.

For their important functions, tunnels in enzymes are also a frequent target of protein engineering efforts. Tunnels, similarly to any voids in proteins, are important hot spots for increasing protein stability. Optimization of the free space and introduction of additional interactions leads to higher increases in the stability compared to mutations in other parts of the protein[86]. On the other hand, because of the tunnel's importance also for the enzyme function, one has to be careful about balancing the stability-activity trade-off. The tunnel lining residues are often important for both properties but the effect of the mutation can be beneficial only for one of them[87]. Regarding functional properties, ligand transport can often be a rate-limiting step of the catalytic cycle[88–90]. Therefore, tunnel lining residues or whole secondary structures around tunnels are often mutated to modify catalytic properties[91]: i) Narrowing of the tunnel may lead to improved catalytic efficiency by constriction of the ligand in the active site[92], ii) tunnel opening accelerate product release or substrate binding[93–95], iii) modification of tunnel length or throughput may lead to change in substrate specificity[96,97], iv) widening of the tunnel bottleneck can create promiscuous enzymes[98], or v) blockage or opening of water tunnel can result in

modification of water flux[99,100]. Modifying all these effects through tunnel mutagenesis is one of the main strategies in protein engineering of enzyme function.

4.2 State-of-the-art

Analysis of the access pathways is challenging, using only experimental techniques. The few available direct methods, such as time-resolved crystallography and crystallography under xenon pressure, are still very time demanding and expensive and provide only limited information[101,102]. Therefore, tunnels and channels can be effectively studied using *in silico* approaches and the field is already well developed[103]. Most of the recent tools, like CAVER 3[104], MOLE 2[105], and MolAxis[106], are based on the pathway detection in the protein structure represented by the Voronoi diagram. These tools, based solely on the analysis of the geometry of access pathways, are high-speed and can provide high-quality results. On the other hand, they do not provide any information about the ligand interaction or energy of the ligand transport.

If one wants to study binding and unbinding processes, the classical experimental approach would be rather indirect, like performing enzyme kinetics experiments to measure the rates or residence times for the ligands[107]. *In silico* methods can be used as a complementary approach for experimental studies. The molecular docking[108–110] is meant to identify optimal binding modes of studied ligands in the binding sites. These methods search for local minima of the binding free energy by perturbating the ligand conformation and evaluating the binding energy by a scoring function. The use of these methods is critical for virtual screenings and drug design[111]. However, molecular docking provides only information about the best binding mode, but it does not take into consideration the transport processes.

Molecular dynamic (MD) simulations are state-of-the-art methods to analyze the motion of protein systems in time and their interactions with ligands[112]. MD simulations can be used to study changes in the protein conformation or binding and unbinding of ligands[92,113]. Unfortunately, ligand transport is often beyond the time limits of classical MD

simulations. Therefore, many enhanced sampling methods were developed to sample larger conformational space[114]. These methods either implement external force or apply different strategies to sample rare events during the simulation. The setup, execution, and analysis of MD simulations require significant knowledge of molecular modeling. To facilitate specific modeling scenarios, tools employing MD, sometimes in the form of user-friendly webservers, were developed[115–118]. Even with a steady improvement of computational power in high-performance computing, using the methods based on MD for large-scale screening studies is not feasible. Therefore, software tools applying approximations to describe the process of ligand transport were recently developed.

Several approximative methods (SLIGHTER[119], MoMA-LigPath[120], ART-RRT[121], GPathFinder[122]) were developed to study interactions between the protein and the ligand during the transport process. These methods, like classical docking, can provide the best binding modes but additionally also provide information on the energetics of the transport and identify bottlenecks limiting the ligand passage. Protein engineering methods can optimize residues in the bottleneck to develop more efficient biocatalysts. These methods use approximations to simulate binding or unbinding fast and provide energy profiles, which can be used to rank ligand or tunnel preferences during the transport process.

4.3 Contribution to the field

Several tools for the analysis of tunnels are available. Unfortunately, most studies focus on a single structure analysis, even though proteins are flexible molecules and only analysis of tunnels in an ensemble of structures can provide a proper description of their geometry. Caver, for example, can handle even analysis of a large number of snapshots from MD simulations but its command-line nature and the necessity of the process automation makes it difficult to use. Therefore, we developed Caver Analyst, a stand-alone tool for quantitative analysis and real-time visualization of tunnels, calculated by Caver, in static structures and molecular simulations. In its second version, Caver Analyst is focused on loading and analyzing even long MD simulation trajectories with advanced analysis and visualization of tunnel dynamics and bottleneck residues. Moreover, Caver Analyst is not

strictly focused only on tunnel analysis. It is mainly a protein visualization tool that enables i) different molecular representations, ii) advanced displaying and coloring techniques, iii) structure alignment, iv) protonation computation, v) measurements, vi) mutagenesis, vii) clip planes and slices, and viii) video recording. The application is supported by the following operating systems: Windows 8 or later, Mac OS X 10.7.5 or later, and major distributions of Linux, including Fedora Core, Red Hat, and Ubuntu. Caver Analyst 2.0 can be downloaded from https://caver.cz and is described in PART II.

Analysis of tunnel geometry, even in MD trajectory, can provide only very limited information about the protein transport processes. Analysis of particular protein-ligand complex and evaluation of the energy of the transport brings an additional level of information. CaverDock is an approximative tool for the identification of possible trajectories of ligand binding or unbinding. The calculation is composed of i) identification of tunnels by Caver, ii) discretization of the tunnel into a series of discs, and iii) constrained docking to each disc employing a modified docking algorithm of AutoDock Vina with parallel heuristics to identify a trajectory of ligand transport. Contrary to MD simulations, the calculation takes only from minutes to a couple of hours, which makes this method applicable also for screening purposes. We tested the screening capabilities of CaverDock in the analysis of inhibitors binding and tunnel preferences in cytochrome P450 and leukotriene A4[123] and the screening of more than 4,300 globally approved drugs binding in Spike protein of SARS-CoV-2[124]. Moreover, CaverDock was also applied by other users to find enantioselective inhibitors[125], explain the mode of inhibition[126], engineering a new catalytic function[127], or identification of hot-spots for mutagenesis[85]. Currently, we are working on Python API for easier setup, calculation, and analysis and on the protein flexibility implementation, which seems to be the major problem in dynamical tunnels. CaverDock can be downloaded from https://loschmidt.chemi.muni.cz/caverdock/ and is thoroughly discussed in PART II.

Both Caver and CaverDock are stand-alone tools that need some basic knowledge in bioinformatics to install and run the calculation correctly. To bring both tools even to

inexperienced users, we have developed a user-friendly web service with a graphical interface called Caver Web. The only mandatory input for tunnel identification and analysis by Caver 3.0 is the protein structure and eventually a list of ligands for the transport analysis by CaverDock 1.0. On the output, the identified tunnels, their properties, lining residues, energy profiles, and trajectories of ligand transport can be visualized. Currently, we are broadening the Caver Web features for an automized virtual screening pipeline of FDA-approved drugs. Moreover, a short molecular dynamic simulation will be introduced into the Caver Web to create an ensemble of structures that can be analysed in a statistically meaningful manner. The server is freely available at https://loschmidt.chemi.muni.cz/caverweb and is thoroughly discussed in PART II.

Access tunnels are functional regions of proteins. Their modification can significantly alter the speed of enzymatic reactions in cases where substrate binding or product release are rate-limiting steps of the catalytic cycle. Therefore, protein engineering of tunnel-lining residues is one of the effective strategies to improve enzyme activity. HotSpot Wizard is a web-based tool for the identification of hot spots for mutagenesis in functional regions, i.e. binding sites or access tunnels. It combines sequence and structure information to identify non-conserved and non-catalytic residues in these functional regions and combines them in smart libraries for screening. In the last version, the HotSpot wizard pipeline (Figure 3) was enriched for prediction and validation of 3D structure from sequence, stability prediction, and recently also molecular docking. HotSpot Wizard 3.0 web server is freely available at https://loschmidt.chemi.muni.cz/hotspotwizard/ and is thoroughly discussed in PART II.

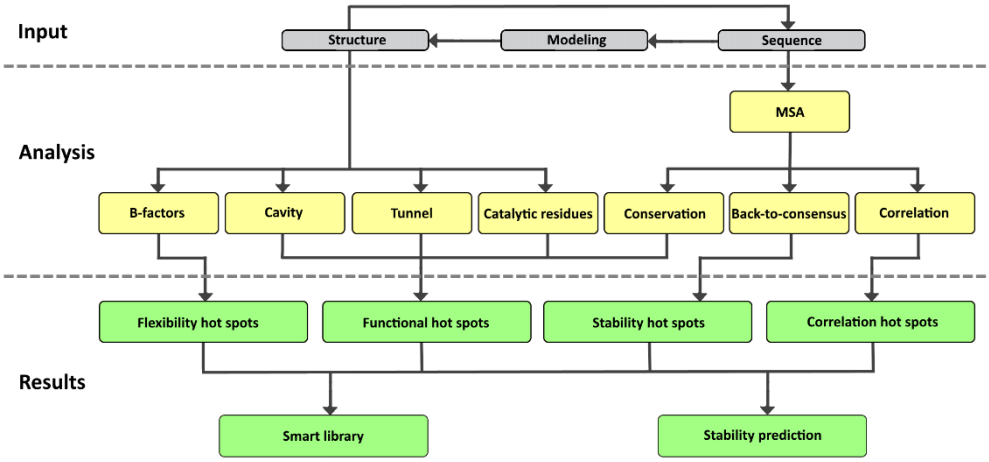

Figure 3. General workflow of HotSpot Wizard. Input files are highlighted in grey, individual analyses in yellow and results on the output in green. Adopted from Planas et al. [/10.1016/j.biotechadv.2021.107696].

REFERENCES

1. Goodwin, S., McPherson, J. D. & McCombie, W. R. Coming of age: ten years of next-generation sequencing technologies. *Nat. Rev. Genet.* **17**, 333–351 (2016).
2. Check Hayden, E. Technology: The $1,000 genome. *Nature* **507**, 294–295 (2014).
3. The UniProt Consortium. UniProt: a hub for protein information. *Nucleic Acids Res.* **43**, D204–D212 (2015).
4. Mitchell, A. L. *et al.* MGnify: the microbiome analysis resource in 2020. *Nucleic Acids Res.* **48**, D570–D578 (2020).
5. Zallot, R., Oberg, N. O. & Gerlt, J. A. 'Democratized' genomic enzymology web tools for functional assignment. *Curr. Opin. Chem. Biol.* **47**, 77–85 (2018).
6. Markel, U. *et al.* Advances in ultrahigh-throughput screening for directed enzyme evolution. *Chem. Soc. Rev.* **49**, 233–262 (2020).
7. Dörr, M. *et al.* Fully automatized high-throughput enzyme library screening using a robotic platform. *Biotechnol. Bioeng.* **113**, 1421–1432 (2016).
8. Colin, P.-Y. *et al.* Ultrahigh-throughput discovery of promiscuous enzymes by picodroplet functional metagenomics. *Nat. Commun.* **6**, 10008 (2015).
9. Mukherjee, S. *et al.* Genomes OnLine Database (GOLD) v.8: overview and updates. *Nucleic Acids Res.* **49**, D723–D733 (2021).
10. Zaparucha, A., Berardinis, V. de & Vaxelaire-Vergne, C. Chapter 1:Genome Mining for Enzyme Discovery. in *Modern Biocatalysis* 1–27 (2018). doi:10.1039/9781788010450-00001.
11. Needleman, S. B. & Wunsch, C. D. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.* **48**, 443–453 (1970).
12. Altschul, S. F., Gish, W., Miller, W., Myers, E. W. & Lipman, D. J. Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410 (1990).
13. Wheeler, D. Selecting the Right Protein-Scoring Matrix. *Curr. Protoc. Bioinforma.* **00**, 3.5.1-3.5.6 (2003).
14. Altschul, S. F. *et al.* Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.* **25**, 3389–3402 (1997).
15. Eddy, S. R. A new generation of homology search tools based on probabilistic inference. *Genome Inform. Int. Conf. Genome Inform.* **23**, 205–211 (2009).
16. Edgar, R. C. Search and clustering orders of magnitude faster than BLAST. *Bioinforma. Oxf. Engl.* **26**, 2460–2461 (2010).
17. Jones, P. *et al.* InterProScan 5: genome-scale protein function classification. *Bioinforma. Oxf. Engl.* **30**, 1236–1240 (2014).
18. Chang, A. *et al.* BRENDA, the ELIXIR core data resource in 2021: new developments and updates. *Nucleic Acids Res.* **49**, D498–D508 (2021).
19. Tsirigos, K. D., Peters, C., Shu, N., Käll, L. & Elofsson, A. The TOPCONS web server for consensus prediction of membrane protein topology and signal peptides. *Nucleic Acids Res.* **43**, W401–W407 (2015).
20. Barrett, T. *et al.* BioProject and BioSample databases at NCBI: facilitating capture and organization of metadata. *Nucleic Acids Res.* **40**, D57–D63 (2012).
21. Jumper, J. *et al.* Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).

22. Vasina, M. *et al.* Tools for computational design and high-throughput screening of therapeutic enzymes. *Adv. Drug Deliv. Rev.* **183**, 114143 (2022).
23. Kramer, R. M., Shende, V. R., Motl, N., Pace, C. N. & Scholtz, J. M. Toward a molecular understanding of protein solubility: increased negative surface charge correlates with increased solubility. *Biophys. J.* **102**, 1907–1915 (2012).
24. Khurana, S. *et al.* DeepSol: a deep learning framework for sequence-based protein solubility prediction. *Bioinformatics* **34**, 2605–2613 (2018).
25. Raimondi, D., Orlando, G., Fariselli, P. & Moreau, Y. Insight into the protein solubility driving forces with neural attention. *PLOS Comput. Biol.* **16**, e1007722 (2020).
26. Trimpin, S. & Brizzard, B. Analysis of insoluble proteins. *BioTechniques* **46**, 409–419 (2009).
27. Helen M. Berman, M. J. G., Andrei Kouranov, David I. Micallef, John Westbrook & investigators, P. S. I. network of. Protein Structure Initiative - TargetTrack 2000-2017 - all data files. (2017) doi:10.5281/zenodo.821654.
28. Price, W. N. *et al.* Large-scale experimental studies show unexpected amino acid effects on protein expression and solubility in vivo in E. coli. *Microb. Inform. Exp.* **1**, 6 (2011).
29. Hirose, S. *et al.* Statistical analysis of features associated with protein expression/solubility in an in vivo Escherichia coli expression system and a wheat germ cell-free expression system. *J. Biochem. (Tokyo)* **150**, 73–81 (2011).
30. Pawlicki, S., Le Béchec, A. & Delamarche, C. AMYPdb: a database dedicated to amyloid precursor proteins. *BMC Bioinformatics* **9**, 273 (2008).
31. Niwa, T. *et al.* Bimodal protein solubility distribution revealed by an aggregation analysis of the entire ensemble of Escherichia coli proteins. *Proc. Natl. Acad. Sci. U. S. A.* **106**, 4201–4206 (2009).
32. Berman, H. M. *et al.* The Protein Data Bank. *Nucleic Acids Res.* **28**, 235–242 (2000).
33. Burley, S. K. *et al.* RCSB Protein Data Bank: powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences. *Nucleic Acids Res.* **49**, D437–D451 (2021).
34. Kawashima, S. & Kanehisa, M. AAindex: amino acid index database. *Nucleic Acids Res.* **28**, 374 (2000).
35. Hon, J. *et al.* SoluProt: Prediction of Soluble Protein Expression in Escherichia coli. *Bioinforma. Oxf. Engl.* btaa1102 (2021) doi:10.1093/bioinformatics/btaa1102.
36. Yang, Y., Niroula, A., Shen, B. & Vihinen, M. PON-Sol: prediction of effects of amino acid substitutions on protein solubility. *Bioinformatics* **32**, 2032–2034 (2016).
37. Sormanni, P., Aprile, F. A. & Vendruscolo, M. The CamSol method of rational design of protein mutants with enhanced solubility. *J. Mol. Biol.* **427**, 478–490 (2015).
38. Tian, Y., Deutsch, C. & Krishnamoorthy, B. Scoring function to predict solubility mutagenesis. *Algorithms Mol. Biol.* **5**, 33 (2010).
39. Paladin, L., Piovesan, D. & Tosatto, S. C. E. SODA: prediction of protein solubility from disorder and aggregation propensity. *Nucleic Acids Res.* **45**, W236–W240 (2017).
40. Van Durme, J. *et al.* Solubis: a webserver to reduce protein aggregation through mutation. *Protein Eng. Des. Sel.* **29**, 285–289 (2016).
41. De Baets, G., Van Durme, J., van der Kant, R., Schymkowitz, J. & Rousseau, F. Solubis: optimize your protein. *Bioinformatics* **31**, 2580–2582 (2015).

42. Zambrano, R. *et al.* AGGRESCAN3D (A3D): server for prediction of aggregation properties of protein structures. *Nucleic Acids Res.* **43**, W306–W313 (2015).
43. Vanacek, P. *et al.* Exploration of Enzyme Diversity by Integrating Bioinformatics with Expression Analysis and Biochemical Characterization. *ACS Catal.* **8**, 2402–2412 (2018).
44. Vanacek, P. *et al.* Functional Annotation of an Enzyme Family by Integrated Strategy Combining Bioinformatics with Microanalytical and Microfluidic Technologies. (2021) doi:10.26434/chemrxiv.13621517.v1.
45. Klesmith, J. R., Bacik, J.-P., Wrenbeck, E. E., Michalczyk, R. & Whitehead, T. A. Trade-offs between enzyme fitness and solubility illuminated by deep mutational scanning. *Proc. Natl. Acad. Sci.* **114**, 2265–2270 (2017).
46. Wrenbeck, E. E. *et al.* An Automated Data-Driven Pipeline for Improving Heterologous Enzyme Expression. *ACS Synth. Biol.* **8**, 474–481 (2019).
47. Bar-Even, A. *et al.* The Moderately Efficient Enzyme: Evolutionary and Physicochemical Trends Shaping Enzyme Parameters. *Biochemistry* **50**, 4402–4410 (2011).
48. Modarres, H. P., Mofrad, M. R. & Sanati-Nezhad, A. Protein thermostability engineering. *RSC Adv.* **6**, 115252–115270 (2016).
49. Pucci, F., Kwasigroch, J. M. & Rooman, M. Protein Thermal Stability Engineering Using HoTMuSiC. *Methods Mol. Biol. Clifton NJ* **2112**, 59–73 (2020).
50. Polizzi, K. M., Bommarius, A. S., Broering, J. M. & Chaparro-Riggers, J. F. Stability of biocatalysts. *Curr. Opin. Chem. Biol.* **11**, 220–225 (2007).
51. Wijma, H. J., Floor, R. J. & Janssen, D. B. Structure- and sequence-analysis inspired engineering of proteins for enhanced thermostability. *Curr. Opin. Struct. Biol.* **23**, 588–594 (2013).
52. Gao, D. *et al.* Thermostable variants of cocaine esterase for long-time protection against cocaine toxicity. *Mol. Pharmacol.* **75**, 318–323 (2009).
53. Goldenzweig, A. *et al.* Automated Structure- and Sequence-Based Design of Proteins for High Bacterial Expression and Stability. *Mol. Cell* **63**, 337–346 (2016).
54. Bloom, J. D., Labthavikul, S. T., Otey, C. R. & Arnold, F. H. Protein stability promotes evolvability. *Proc. Natl. Acad. Sci. U. S. A.* **103**, 5869–5874 (2006).
55. Kretz, K. A. *et al.* Gene site saturation mutagenesis: a comprehensive mutagenesis approach. *Methods Enzymol.* **388**, 3–11 (2004).
56. Seitz, T. *et al.* Enhancing the stability and solubility of the glucocorticoid receptor ligand-binding domain by high-throughput library screening. *J. Mol. Biol.* **403**, 562–577 (2010).
57. Bommarius, A. S. & Paye, M. F. Stabilizing biocatalysts. *Chem. Soc. Rev.* **42**, 6534–6565 (2013).
58. Sumbalova, L., Stourac, J., Martinek, T., Bednar, D. & Damborsky, J. HotSpot Wizard 3.0: web server for automated design of mutations and smart libraries based on sequence input information. *Nucleic Acids Res.* **46**, W356–W362 (2018).
59. Wijma, H. J., Fürst, M. J. L. J. & Janssen, D. B. A Computational Library Design Protocol for Rapid Improvement of Protein Stability: FRESCO. *Methods Mol. Biol. Clifton NJ* **1685**, 69–85 (2018).
60. Kellogg, E. H., Leaver-Fay, A. & Baker, D. Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins Struct. Funct. Bioinforma.* **79**, 830–838 (2011).

61. Guerois, R., Nielsen, J. E. & Serrano, L. Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J. Mol. Biol.* **320**, 369–387 (2002).
62. Schymkowitz, J. *et al.* The FoldX web server: an online force field. *Nucleic Acids Res.* **33**, W382-388 (2005).
63. Yin, S., Ding, F. & Dokholyan, N. V. Eris: an automated estimator of protein stability. *Nat. Methods* **4**, 466–467 (2007).
64. Magliery, T. J. Protein stability: computation, sequence statistics, and new experimental methods. *Curr. Opin. Struct. Biol.* **33**, 161–168 (2015).
65. Watanabe, K., Ohkuri, T., Yokobori, S. & Yamagishi, A. Designing thermostable proteins: ancestral mutants of 3-isopropylmalate dehydrogenase designed by using a phylogenetic tree. *J. Mol. Biol.* **355**, 664–674 (2006).
66. Wheeler, L. C., Lim, S. A., Marqusee, S. & Harms, M. J. The thermostability and specificity of ancient proteins. *Curr. Opin. Struct. Biol.* **38**, 37–43 (2016).
67. Chaloupkova, R. *et al.* Light-Emitting Dehalogenases: Reconstruction of Multifunctional Biocatalysts. *ACS Catal.* **9**, 4810–4823 (2019).
68. Babkova, P. *et al.* Structures of hyperstable ancestral haloalkane dehalogenases show restricted conformational dynamics. *Comput. Struct. Biotechnol. J.* **18**, 1497–1508 (2020).
69. Nikam, R., Kulandaisamy, A., Harini, K., Sharma, D. & Gromiha, M. M. ProThermDB: thermodynamic database for proteins and mutants revisited after 15 years. *Nucleic Acids Res.* **49**, D420–D424 (2021).
70. Pucci, F., Bernaerts, K. V., Kwasigroch, J. M. & Rooman, M. Quantification of biases in predictions of protein stability changes upon mutations. *Bioinforma. Oxf. Engl.* **34**, 3659–3665 (2018).
71. Mazurenko, S. Predicting protein stability and solubility changes upon mutations: data perspective. *ChemCatChem* **12**, 5590–5598 (2020).
72. Dehouck, Y. *et al.* Fast and accurate predictions of protein stability changes upon mutations using statistical potentials and neural networks: PoPMuSiC-2.0. *Bioinformatics* **25**, 2537–2543 (2009).
73. Khan, S. & Vihinen, M. Performance of protein stability predictors. *Hum. Mutat.* **31**, 675–684 (2010).
74. Potapov, V., Cohen, M. & Schreiber, G. Assessing computational methods for predicting protein stability upon mutation: good on average but not in the details. *Protein Eng. Des. Sel.* **22**, 553–560 (2009).
75. Musil, M., Konegger, H., Hon, J., Bednar, D. & Damborsky, J. Computational Design of Stable and Soluble Biocatalysts. *ACS Catal.* **9**, 1033–1054 (2019).
76. Wijma, H. J. *et al.* Computationally designed libraries for rapid enzyme stabilization. *Protein Eng. Des. Sel.* **27**, 49–58 (2014).
77. Bednar, D. *et al.* FireProt: Energy- and Evolution-Based Computational Design of Thermostable Multiple-Point Mutants. *PLOS Comput. Biol.* **11**, e1004556 (2015).
78. Xia, Y. *et al.* Development of thermostable sucrose phosphorylase by semi-rational design for efficient biosynthesis of alpha-D-glucosylglycerol. *Appl. Microbiol. Biotechnol.* **105**, 7309–7319 (2021).
79. Liu, Y. *et al.* Enhancing the thermal stability of ketoreductase ChKRED12 using the FireProt web server. *Process Biochem.* **101**, 207–212 (2021).
80. Cheng, Z. *et al.* Computational Design of Nitrile Hydratase from Pseudonocardia thermophila JCM3095 for Improved Thermostability. *Molecules* **25**, 4806 (2020).

81. Solarczek, J. *et al.* Position 123 of halohydrin dehalogenase HheG plays an important role in stability, activity, and enantioselectivity. *Sci. Rep.* **9**, 5106 (2019).
82. Pravda, L. *et al.* Anatomy of enzyme channels. *BMC Bioinformatics* **15**, 379 (2014).
83. Kokkonen, P. *et al.* Substrate inhibition by the blockage of product release and its control by tunnel engineering. *RSC Chem. Biol.* **2**, 645–655 (2021).
84. Marques, S. M., Bednar, D. & Damborsky, J. Computational Study of Protein-Ligand Unbinding for Enzyme Engineering. *Front. Chem.* **6**, (2019).
85. Rapp, L. R. *et al.* Substrate Anchoring and Flexibility Reduction in CYP153AM.aq Leads to Highly Improved Efficiency toward Octanoic Acid. *ACS Catal.* **11**, 3182–3189 (2021).
86. Koudelakova, T. *et al.* Engineering Enzyme Stability and Resistance to an Organic Cosolvent by Modification of Residues in the Access Tunnel. *Angew. Chem. Int. Ed.* **52**, 1959–1963 (2013).
87. Liskova, V. *et al.* Balancing the Stability–Activity Trade-Off by Fine-Tuning Dehalogenase Access Tunnels. *ChemCatChem* **7**, 648–659 (2015).
88. Kokkonen, P. *et al.* The impact of tunnel mutations on enzymatic catalysis depends on the tunnel-substrate complementarity and the rate-limiting step. *Comput. Struct. Biotechnol. J.* **18**, 805–813 (2020).
89. Wang, L. H., Tsai, A. L. & Hsu, P. Y. Substrate binding is the rate-limiting step in thromboxane synthase catalysis. *J. Biol. Chem.* **276**, 14737–14743 (2001).
90. Shannon, A. E. *et al.* Product release is rate-limiting for catalytic processing by the Dengue virus protease. *Sci. Rep.* **6**, 37539 (2016).
91. Schenkmayerova, A. *et al.* Engineering the protein dynamics of an ancestral luciferase. *Nat. Commun.* **12**, 3616 (2021).
92. Marques, S. M. *et al.* Catalytic Cycle of Haloalkane Dehalogenases Toward Unnatural Substrates Explored by Computational Modeling. *J. Chem. Inf. Model.* **57**, 1970–1989 (2017).
93. Hamre, A. G., Frøberg, E. E., Eijsink, V. G. H. & Sørlie, M. Thermodynamics of tunnel formation upon substrate binding in a processive glycoside hydrolase. *Arch. Biochem. Biophys.* **620**, 35–42 (2017).
94. Brezovsky, J. *et al.* Engineering a de Novo Transport Tunnel. *ACS Catal.* **6**, 7597–7610 (2016).
95. Kong, X.-D. *et al.* Engineering of an epoxide hydrolase for efficient bioresolution of bulky pharmaco substrates. *Proc. Natl. Acad. Sci.* **111**, 15717–15722 (2014).
96. Subramanian, K. *et al.* Modulating D-amino acid oxidase (DAAO) substrate specificity through facilitated solvent access. *PLOS ONE* **13**, e0198990 (2018).
97. Finzel, K. *et al.* Probing the Substrate Specificity and Protein-Protein Interactions of the E. coli Fatty Acid Dehydratase, FabA. *Chem. Biol.* **22**, 1453–1460 (2015).
98. Yan, X., Wang, J., Sun, Y., Zhu, J. & Wu, S. Facilitating the Evolution of Esterase Activity from a Promiscuous Enzyme (Mhg) with Catalytic Functions of Amide Hydrolysis and Carboxylic Acid Perhydrolysis by Engineering the Substrate Entrance Tunnel. *Appl. Environ. Microbiol.* **82**, 6748–6756 (2016).
99. Syrén, P.-O., Hammer, S. C., Claasen, B. & Hauer, B. Entropy is key to the formation of pentacyclic terpenoids by enzyme-catalyzed polycyclization. *Angew. Chem. Int. Ed Engl.* **53**, 4845–4849 (2014).
100. David, B. *et al.* Internal Water Dynamics Control the Transglycosylation/Hydrolysis Balance in the Agarase (AgaD) of Zobellia galactanivorans. *ACS Catal.* **7**, 3357–3367 (2017).

101. Šrajer, V. *et al.* Protein Conformational Relaxation and Ligand Migration in Myoglobin: A Nanosecond to Millisecond Molecular Movie from Time-Resolved Laue X-ray Diffraction. *Biochemistry* **40**, 13802–13815 (2001).
102. Schmidt, M. *et al.* Ligand migration pathway and protein dynamics in myoglobin: A time-resolved crystallographic study on L29W MbCO. *Proc. Natl. Acad. Sci.* **102**, 11704–11709 (2005).
103. Brezovsky, J. *et al.* Software tools for identification, visualization and analysis of protein tunnels and channels. *Biotechnol. Adv.* **31**, 38–49 (2013).
104. Chovancova, E. *et al.* CAVER 3.0: A Tool for the Analysis of Transport Pathways in Dynamic Protein Structures. *PLOS Comput. Biol.* **8**, e1002708 (2012).
105. Sehnal, D. *et al.* MOLE 2.0: advanced approach for analysis of biomacromolecular channels. *J. Cheminformatics* **5**, 39 (2013).
106. Yaffe, E., Fishelovitch, D., Wolfson, H. J., Halperin, D. & Nussinov, R. MolAxis: a server for identification of channels in macromolecules. *Nucleic Acids Res.* **36**, W210–W215 (2008).
107. Schuetz, D. A. *et al.* Kinetics for Drug Discovery: an industry-driven effort to target drug residence time. *Drug Discov. Today* **22**, 896–911 (2017).
108. Trott, O. & Olson, A. J. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem.* **31**, 455–461 (2010).
109. Morris, G. M. *et al.* AutoDock4 and AutoDockTools4: Automated docking with selective receptor flexibility. *J. Comput. Chem.* **30**, 2785–2791 (2009).
110. Verdonk, M. L., Cole, J. C., Hartshorn, M. J., Murray, C. W. & Taylor, R. D. Improved protein-ligand docking using GOLD. *Proteins* **52**, 609–623 (2003).
111. Kitchen, D. B., Decornez, H., Furr, J. R. & Bajorath, J. Docking and scoring in virtual screening for drug discovery: methods and applications. *Nat. Rev. Drug Discov.* **3**, 935–949 (2004).
112. Hospital, A., Goñi, J. R., Orozco, M. & Gelpi, J. L. Molecular dynamics simulations: advances and applications. *Adv. Appl. Bioinforma. Chem.* **8**, 37–47 (2015).
113. Kokkonen, P. *et al.* Molecular Gating of an Engineered Enzyme Captured in Real Time. *J. Am. Chem. Soc.* **140**, 17999–18008 (2018).
114. Rydzewski, J. & Nowak, W. Ligand diffusion in proteins via enhanced sampling in molecular dynamics. *Phys. Life Rev.* **22–23**, 58–74 (2017).
115. Bruce, N. J., Ganotra, G. K., Richter, S. & Wade, R. C. KBbox: A Toolbox of Computational Methods for Studying the Kinetics of Molecular Binding. *J. Chem. Inf. Model.* **59**, 3630–3634 (2019).
116. Kingsley, L. J. & Lill, M. A. Including ligand-induced protein flexibility into protein tunnel prediction. *J. Comput. Chem.* **35**, 1748–1756 (2014).
117. Yang, J.-F., Wang, F., Chen, Y.-Z., Hao, G.-F. & Yang, G.-F. LARMD: integration of bioinformatic resources to profile ligand-driven protein dynamics with a case on the activation of estrogen receptor. *Brief. Bioinform.* **21**, 2206–2218 (2020).
118. Stank, A. *et al.* TRAPP webserver: predicting protein binding site flexibility and detecting transient binding pockets. *Nucleic Acids Res.* **45**, W325–W330 (2017).

119. Lee, P.-H., Kuo, K.-L., Chu, P.-Y., Liu, E. M. & Lin, J.-H. SLITHER: a web server for generating contiguous conformations of substrate molecules entering into deep active sites of proteins or migrating through channels in membrane transporters. *Nucleic Acids Res.* **37**, W559–W564 (2009).
120. Devaurs, D. *et al.* MoMA-LigPath: a web server to simulate protein-ligand unbinding. *Nucleic Acids Res.* **41**, W297-302 (2013).
121. Nguyen, M. K., Jaillet, L. & Redon, S. ART-RRT: As-Rigid-As-Possible exploration of ligand unbinding pathways. *J. Comput. Chem.* **39**, 665–678 (2018).
122. Sánchez-Aparicio, J.-E. *et al.* GPathFinder: Identification of Ligand-Binding Pathways by a Multi-Objective Genetic Algorithm. *Int. J. Mol. Sci.* **20**, 3155 (2019).
123. Pinto, G. P. *et al.* Fast Screening of Inhibitor Binding/Unbinding Using Novel Software Tool CaverDock. *Front. Chem.* **7**, (2019).
124. Pinto, G. P. *et al.* Screening of world approved drugs against highly dynamical spike glycoprotein of SARS-CoV-2 using CaverDock and machine learning. *Comput. Struct. Biotechnol. J.* **19**, 3187–3197 (2021).
125. Knez, D. *et al.* Stereoselective Activity of 1-Propargyl-4-styrylpiperidine-like Analogues That Can Discriminate between Monoamine Oxidase Isoforms A and B. *J. Med. Chem.* **63**, 1361–1387 (2020).
126. Singh, P. P., Jaiswal, A. K., Kumar, A., Gupta, V. & Prakash, B. Untangling the multi-regime molecular mechanism of verbenol-chemotype Zingiber officinale essential oil against Aspergillus flavus and aflatoxin B1. *Sci. Rep.* **11**, 6832 (2021).
127. Papadopoulou, A. *et al.* Re-Programming and Optimization of a L-Proline cis-4-Hydroxylase for the cis-3-Halogenation of its Native Substrate. *ChemCatChem* **13**, 3914–3919 (2021).

AUTHOR CONTRIBUTION

I declare that my contribution to the publications was as:

1. EnzymeMiner – workflow design, graphical interface design, supervision, testing, writing of the manuscript
2. SoluProt - workflow design, supervision, writing of the manuscript
3. FireProt – conceptualization, workflow design, graphical interface design, supervision, testing, writing of the manuscript
4. FireProtASR – conceptualization, workflow design, graphical interface design, supervision, testing, writing of the manuscript
5. FireProtDB – conceptualization, data curation, graphical interface design, supervision, testing, writing of the manuscript
6. Caver Analyst 2 - conceptualization, testing, writing of the manuscript
7. CaverDock - conceptualization, supervision, writing of the manuscript
8. CaverWeb - conceptualization, workflow design, graphical interface design, supervision, testing, writing of the manuscript
9. HotSpot Wizard - conceptualization, supervision, testing, writing of the manuscript

PART II

SELECTED PUBLICATIONS

EnzymeMiner: Automated Mining of Soluble Enzymes with Diverse Structures, Catalytic Properties and Stabilities.

W104–W109 *Nucleic Acids Research, 2020, Vol. 48, Web Server issue* *Published online 11 May 2020*
doi: 10.1093/nar/gkaa372

EnzymeMiner: automated mining of soluble enzymes with diverse structures, catalytic properties and stabilities

Jiri Hon[1,2,3,†], Simeon Borko[1,2,†], Jan Stourac[1,3], Zbynek Prokop[1,3], Jaroslav Zendulka[2], David Bednar[1,3], Tomas Martinek[2] and Jiri Damborsky[1,3,*]

[1]Loschmidt Laboratories, Department of Experimental Biology and Research Center for Toxic Compounds in the Environment RECETOX, Faculty of Science, Masaryk University, Brno, Czech Republic, [2]IT4Innovations Centre of Excellence, Faculty of Information Technology, Brno University of Technology, Bozetechova 2, Brno, Czech Republic and [3]International Clinical Research Center, St. Anne's University Hospital Brno, Brno, Czech Republic

Received March 08, 2020; Revised April 13, 2020; Editorial Decision April 27, 2020; Accepted April 29, 2020

ABSTRACT

Millions of protein sequences are being discovered at an incredible pace, representing an inexhaustible source of biocatalysts. Despite genomic databases growing exponentially, classical biochemical characterization techniques are time-demanding, cost-ineffective and low-throughput. Therefore, computational methods are being developed to explore the unmapped sequence space efficiently. Selection of putative enzymes for biochemical characterization based on rational and robust analysis of all available sequences remains an unsolved problem. To address this challenge, we have developed EnzymeMiner—a web server for automated screening and annotation of diverse family members that enables selection of hits for wet-lab experiments. EnzymeMiner prioritizes sequences that are more likely to preserve the catalytic activity and are heterologously expressible in a soluble form in *Escherichia coli*. The solubility prediction employs the in-house SoluProt predictor developed using machine learning. EnzymeMiner reduces the time devoted to data gathering, multi-step analysis, sequence prioritization and selection from days to hours. The successful use case for the haloalkane dehalogenase family is described in a comprehensive tutorial available on the EnzymeMiner web page. EnzymeMiner is a universal tool applicable to any enzyme family that provides an interactive and easy-to-use web interface freely available at https://loschmidt.chemi.muni.cz/enzymeminer/.

INTRODUCTION

There are currently >259 million non-redundant protein sequences in the NCBI nr database (release 2020-02-10) (1). Despite their enormous promise for biological and biotechnological discovery, experimental characterization has been performed on only a small fraction of the available sequences. Currently, there are about 560 000 protein sequences reliably curated in the UniProtKB/Swiss-Prot database (release 2020_01) (2).

The low ratio of characterized to uncharacterized sequences reflects the sharp contrast in time-demanding/low-throughput biochemical techniques versus fast/high-throughput next-generation sequencing technology. Although more efficient biochemical techniques employing miniaturization and automation have been developed (3–5), the most widely used experimental methods do not provide sufficient capacity for biochemical characterization of proteins spanning the ever-increasing sequence space. Therefore, computational methods are currently the only way to explore the immense protein diversity available among the millions of uncharacterized sequence entries.

Two different computational strategies are generally used for exploration of the unknown sequence space. The first strategy takes a novel uncharacterized sequence as input and predicts functional annotations. The method involves annotating the unknown input sequences by predicting protein domains (6), Enzyme Commission (EC) number (7) or Gene Ontology terms that are a subject of the initiative named the Critical Assessment of Functional Annotation (8). These methods are often universal and applicable to any protein sequence. However, they often lack specificity as the automatic annotation rules or statistical models need to be substantially general. A significant advantage of these methods is their seamless integration into available

*To whom correspondence should be addressed. Tel: +420 5 4949 3467; Email: jiri@chemi.muni.cz
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

databases. Submission of a query sequence to a database is sufficient, with no need for running computation- and memory-intensive bioinformatics pipelines locally. A model example of this approach is the automatic annotation workflow of the UniProtKB/TrEMBL database (2).

The second strategy takes a well-known characterized sequence as an input and applies a computational workflow, typically based on a homology search, to identify novel uncharacterized entries in genomic databases that are related to the input query sequence (5,9). The homology search is often followed by a filtration step, which checks the essential sequence properties, e.g. domain structure or presence of catalytic residues. The main advantage of these methods is the higher specificity of the analysis. A disadvantage is that it may be complicated to apply the developed workflow to protein families other than those for which it was designed. Moreover, these workflows typically require running complex bioinformatics pipelines and are usually not available through a web interface.

The fundamental unsolved problem is how to deal with the overwhelming number of sequence entries identified by these methods and select a small number of relevant hits for in-depth experimental characterization. For example, a database search for members of the haloalkane dehalogenase model family using the UniProt web interface yields 3598 sequences (UniProtKB release 2020_01). It is impossible to rationally select several tens of targets for experimental testing without additional bioinformatics analyses to help prioritize such a large pool of sequences.

To address the challenge of exploring the unmapped enzyme sequence space and rational selection of attractive targets, we have developed the EnzymeMiner web server. EnzymeMiner identifies novel enzyme family members, comprehensively annotates the targets and facilitates efficient prioritization and selection of representative hits for experimental characterization. To the best of our knowledge, there is currently no other tool available that allows such a comprehensive analysis in a single easy-to-run integrated workflow on the web.

MATERIALS AND METHODS

EnzymeMiner implements a three-step workflow: (i) homology search, (ii) essential residue based filtering and (iii) hits annotation (Figure 1). To execute these tasks, the server requires two different types of input information: (i) query sequences and (ii) essential residue templates. The query sequences serve as seeds for the initial homology search. The essential residue templates, defined as pairs of a protein sequence and a set of essential residues in that sequence, allow the server to prioritize hits that are more likely to display the enzyme function. Therefore, the essential residues may be the catalytic and ligand- or cofactor-binding residues that are indispensable for proper catalytic function. Each essential residue is defined by its name, position and a set of allowed amino acids for that position.

In the first *homology search step*, a query sequence is used as a query for a PSI-BLAST (10) two-iteration search in the NCBI nr database (1). If more than one query sequence is provided, a search is conducted for each sequence separately. Besides a minimum *E*-value threshold 10^{-20}, the PSI-BLAST hits must share a minimum of 25% global sequence identity with at least one of the query sequences. Artificial protein sequences, i.e. sequences described by the term artificial, synthetic construct, vector, vaccinia virus, plasmid, halotag or replicon, are removed. EnzymeMiner sorts the PSI-BLAST hits by *E*-value and passes a maximum of 10,000 best hits to the next steps in the workflow. The default parameters for the homology search step, as well as the other steps, can be modified using advanced options in the web server.

In the second *essential residue based filtering step*, the homology search hits are filtered using the essential residue templates. First, the hits are divided into template clusters. Each cluster contains all hits matching essential residues of a particular template. Essential residues are checked using global pairwise alignment with the template calculated by USEARCH (11). When multiple essential residue templates match, the hit is assigned to the template with the highest global sequence identity. Second, for each cluster, an initial multiple sequence alignment (MSA) is constructed using Clustal Omega (12). The MSA is used to revalidate the essential residues of identified hits by checking the corresponding column in the MSA. Sequences not matching essential residues of the template are removed from the cluster. Third, the MSA is constructed again for each template cluster and the essential residues are checked for the last time. The final set of identified sequences reported by EnzymeMiner contains all sequences left in the template clusters.

In the third *annotation step*, the identified sequences are annotated using several databases and predictors: (i) transmembrane regions are predicted by TMHMM (13), (ii) Pfam domains are predicted by InterProScan (14), (iii) source organism annotation is extracted from the NCBI Taxonomy (15) and the NCBI BioProject database (16), (iv) protein solubility is predicted by the in-house tool SoluProt for prediction of soluble protein expression in *Escherichia coli* and (v) sequence identities to queries, hits or other optional sequences are calculated by USEARCH (11). SoluProt is based on a random forest regression model that employs 36 sequence-based features (https://loschmidt.chemi.muni.cz/soluprot/). It has been shown to achieve an accuracy of 58%, specificity of 73% and sensitivity of 44% on a balanced independent test set of 3788 sequences (Hon et al., manuscript in preparation). Alternative solubility prediction tools are summarised in a recently published review (17). It is not advised to use the solubility score for other expression systems because it was trained solely on *E. coli* data. We expect further intensive development of protein solubility predictors in coming years and will ensure that the solubility score in the EnzymeMiner stays at the cutting-edge in terms of its accuracy and reproducibility.

The sequence space of the identified hits is visualized using representative sequence similarity networks (SSNs) generated at various clustering thresholds using MMseqs2 (18) and Cytoscape (19). SSNs provide a clean visual approach to identify clusters of highly similar sequences and rapidly spot sequence outliers. SSNs proved to facilitate identification of previously unexplored sequence and function space (20). The SSN generation method used in EnzymeMiner is inspired by the EFI-EST tool (21). The minimum align-

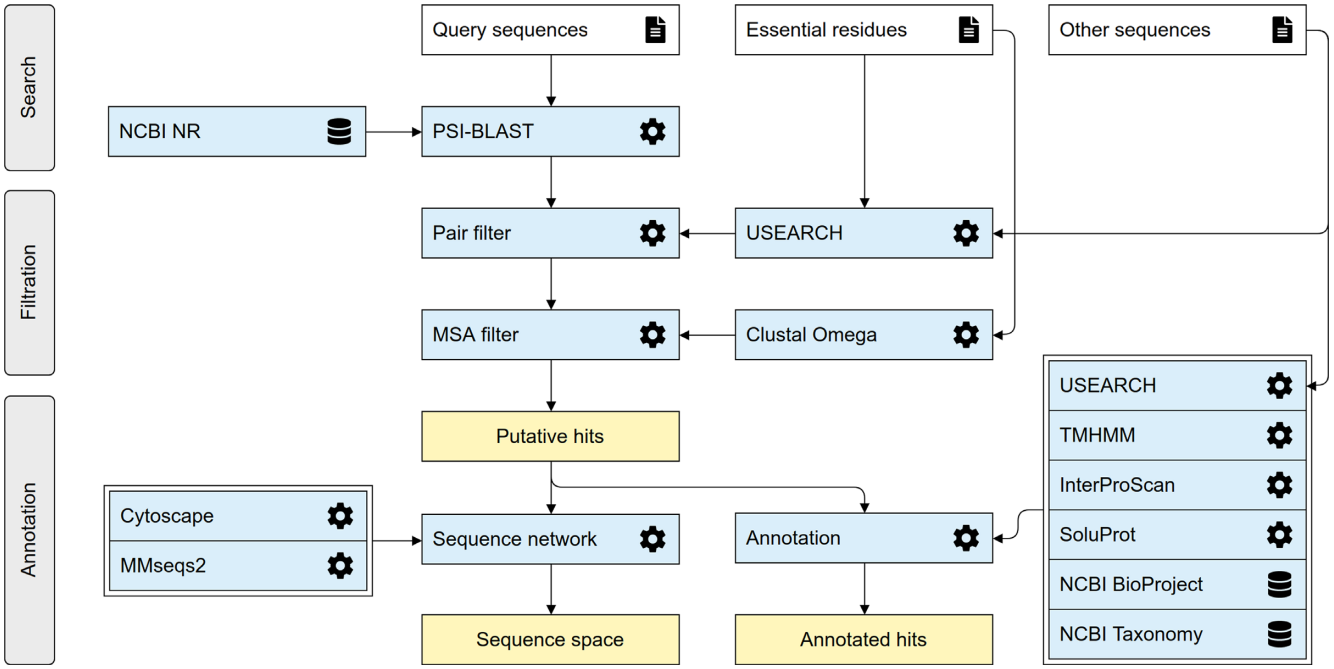

Figure 1. The EnzymeMiner workflow. The workflow consists of three distinct steps: (i) sequence homology search, (ii) filtration of functional sequences, and (iii) annotation of hits. These steps are executed consecutively and automatically. EnzymeMiner has only two required inputs: (i) query sequences, and (ii) essential residue templates. The *Other sequences* are optional inputs that allow EnzymeMiner to calculate the sequence identity between these sequences and all the hits. Input files are highlighted by a white background, tools and databases have a light blue background, outputs are highlighted by a yellow background.

ment score to include an edge between two representative sequences in an SSN is 40.

DESCRIPTION OF THE WEB SERVER

Job submission

New jobs can be submitted from the EnzymeMiner homepage. EnzymeMiner provides two conceptually different ways to define the input of the workflow: (i) using curated sequences from the UniProtKB/Swiss-Prot database and (ii) using custom sequences. We recommend the UniProtKB/Swiss-Prot option for users who do not have in-depth knowledge of the enzyme family. In contrast, the *Custom sequences* tab gives full control over the EnzymeMiner input—query sequences and essential residue templates are specified manually by the user. This is recommended for users who have good knowledge about the enzyme family and want to provide additional starting information to obtain refined results. The last option is a combination of both approaches, where Swiss-Prot sequences can be pre-selected first and then the input can be modified in the *Custom sequences* tab.

In the *Swiss-Prot sequences* tab (Figure 2A), sequences from the Swiss-Prot database can be queried by Enzyme Commission (EC) number. As a result, a table of all sequences annotated by the EC number and corresponding SSN is generated. The table has four columns: (i) sequence accessions hyperlinked to the UniProt database, (ii) number of essential residues, (iii) sequence length and (iv) sequence plot. The sequence plot summarizes two important features of the sequence – positions of essential residues and identified Pfam domains. The positions of essential residues are obtained from the Swiss-Prot database. The SSN visualizes the sequence space of all the sequences in the current EC group. Nodes represent Swiss-Prot sequences, whereas edge lengths are proportional to the pairwise sequence identities. Similar sequences are close to each other, whereas more distant sequences are not connected at all.

There are three strategies possible for selecting Swiss-Prot sequences as the EnzymeMiner query: (i) select a row from the sequence table, (ii) select a node in the SSN and (iii) select cluster representatives by defining a sequence identity threshold. The sequence identity threshold buttons select cluster representatives at the given percentage threshold. Using this feature, the user can automatically select a small set of sequences that cover the whole known sequence space of the current EC group. All selected Swiss-Prot sequences are used as a query in the homology search step and also as essential residue templates for the filtration step. To modify the selected sets of queries and essential residue templates, the user can switch to the *Custom sequences* tab and refine the selection manually.

EnzymeMiner results

The results page is organized into four sections: (i) *job information* box, (ii) *download results* box, (iii) *target selection table* and (iv) *sequence similarity network*.

In the *job information* box, the user can find the job ID, title, start time and status of the job. There is also a rerun button for rerunning the same analysis without the need for re-entering the same input. This feature is handy for periodically mining new sequences as the sequence databases

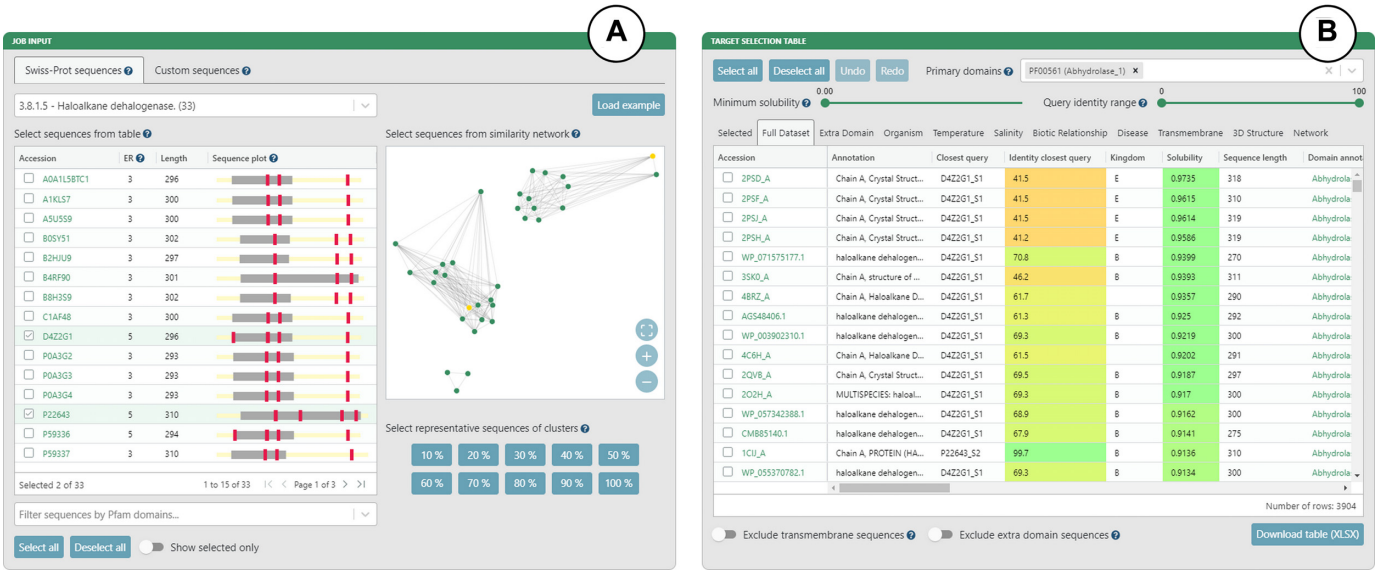

Figure 2. The EnzymeMiner graphical user interface showing example inputs and results for the haloalkane dehalogenase family (EC 3.8.1.5). (**A**) Inputs based on curated sequences from the UniProtKB/Swiss-Prot database. The input sequences can be selected using: (i) the sequence table, (ii) the SSN or (iii) the sequence identity threshold. (**B**) Target selection table. The table is organized into eleven sheets that summarize the results from different perspectives. The table can be filtered using solubility and identity sliders, and transmembrane and extra domain exclusion switches.

grow. For example, there are hundreds of new hits for the haloalkane dehalogenase family every year. In the *download results* box, the user can download the results table in XLSX format or tab-separated text format. A ZIP archive containing all output files from the EnzymeMiner workflow can also be downloaded.

The *target selection table* is the most important component of the EnzymeMiner results (Figure 2B). It presents all the putative enzyme sequences identified by EnzymeMiner and helps to select targets for experimental characterization. The table is organized into eleven sheets summarizing the results from different perspectives. (i) The *Selected* sheet shows all the sequences selected from individual sheets. It contains an extra column to track the argument used for the selection. By default, it is prefilled by the name of the sheet from which the sequence was selected, but it can be freely changed. (ii) The *Full Dataset* sheet shows all identified sequences. (iii) The *Extra domain* sheet shows sequences with extra Pfam domains found in the sequence but not listed in the *Primary domains* selection box. (iv) The *Organism* sheet shows sequences with known source organisms. (v) The *Temperature* sheet shows sequences from organisms having extreme optimum temperature annotation in the NCBI BioProject database, including sequences from thermophilic or cryophilic organisms. (vi) The *Salinity* sheet shows sequences from organisms having extreme salinity annotation in the NCBI BioProject database. (vii) The *Biotic Relationship* sheet shows sequences from organisms having biotic relationship annotation in the NCBI BioProject database. (viii) The *Disease* sheet shows sequences from organisms having disease annotation in the NCBI BioProject database. (ix) The *Transmembrane* sheet shows sequences with transmembrane regions predicted by the TMHMM tool. (x) The *3D Structure* sheet shows sequences with an available 3D structure in the Protein Data Bank (22). (xi) The *Network* sheet shows sequences clustered into a selected sequence similarity network node.

There are four options for filtering the identified sequences displayed in the target selection table. The first option is the minimum solubility slider. Sequences with lower predicted solubility will be hidden. We recommend setting the solubility threshold to >0.5 to increase the probability of finding soluble protein expression in *E. coli*. We do not recommend to set the solubility threshold too high because of possible trade-off between enzyme solubility and activity (23). The second option is the identity range bar. Only sequences with identity to query sequences in the specified range will be visible. The third option is to exclude transmembrane proteins. We recommend removing these sequences as they are usually difficult to produce and tend to have lower predicted solubility. The fourth option is to exclude proteins with an extra domain. Extra domains are defined as domains found in the sequence but not listed in the *Primary domains* selection box. We recommend avoiding sequences with extra domains, but these sequences may also show interesting and unusual biological properties. The selection table can be sorted by clicking on a column header. Holding 'Shift' while clicking on the column headers allows sorting by multiple columns.

The SSN visualizes the sequence space of all identified sequences. Both clusters of similar sequences and sequence outliers can be easily identified. As there might be thousands of sequences, the sequences are clustered at the identity threshold and only an SSN of the representative sequences is shown for performance reasons. Sequences having greater sequence identity are consolidated into a single metanode. Edges indicate high sequence identity between representative sequences of the connected metanodes. Clicking on a metanode displays the *Network* sheet showing

which sequences are represented by a particular metanode. The SSN can be downloaded as a Cytoscape session file for further analysis and custom visualization. Networks clustered at different identities are available. The numbers of nodes and edges are indicated for each identity threshold. The SSN is interactively linked to the target selection table. All nodes representing selected sequences are automatically highlighted in the SSN.

Target selection

The target selection table and SSN facilitate the selection of a diverse set of soluble putative enzyme sequences for experimental validation. First, we recommend setting the maximum sequence identity to queries to 90%. This will remove all hits that are very similar to already known proteins. Second, we recommend selecting a few sequences from individual sheets to cover different phyla from the domains Archea, Bacteria and Eukarya. The most exciting enzymes might be from extremophilic organisms. Third, the SSN can be used to check that the selection covers all sequence clusters. Fourth, users can select sequences from all subfamilies of the enzyme family of interest. The members of different subfamilies can be easily recognized by the *Closest query* or *Closest known* column in the selection table (note: requires representative sequences of subfamilies as job input). Fifth, the available filtering options can be used to (i) prioritize sequences with the highest predicted solubility, (ii) prioritize sequences with known tertiary structures, (iii) eliminate proteins with predicted transmembrane regions and (iv) eliminate sequences with extra domains.

EXPERIMENTAL VALIDATION OF THE EnzymeMiner WORKFLOW

The EnzymeMiner workflow has been thoroughly experimentally validated using the model enzymes haloalkane dehalogenases (5). The sequence-based search identified 658 putative dehalogenases. The subsequent analysis prioritized and selected 20 candidate genes for exploration of their protein structural and functional diversity. The selected enzymes originated from genetically unrelated Bacteria, Eukarya and, for the first time, also Archaea and showed novel catalytic properties and stabilities. The workflow helped to identify novel haloalkane dehalogenases, including (i) the most catalytically efficient enzyme ($k_{cat}/K_{0.5} = 96.8\ mM^{-1}\ s^{-1}$), (ii) the most thermostable enzyme showing a melting temperature of 71°C, (iii) three different cold-adapted enzymes active at near to 0°C, (iv) highly enantioselective enzymes, (v) enzymes with a wide range of optimal operational temperature from 20 to 70°C and an unusually broad pH range from 5.7–10 and (vi) biocatalysts degrading the warfare chemical yperite and various environmental pollutants. The sequence mining, annotation, and visualization steps from the workflow published by Vanacek and coworkers (5) were fully automated in the EnzymeMiner web server. The successful use case for the haloalkane dehalogenase family is described in an easy-to-follow tutorial available on the EnzymeMiner web page. Additional extensive validation of the fully automated version of EnzymeMiner, experimentally testing the properties of another 45 genes of the haloalkane dehalogenases, is currently ongoing in our laboratory.

CONCLUSIONS AND OUTLOOK

The EnzymeMiner web server identifies putative members of enzyme families and facilitates their prioritization and well-informed manual selection for experimental characterization to reveal novel biocatalysts. Such a task is difficult using the web interfaces of the available protein databases, e.g. UniProtKB/TrEMBL and NCBI Protein, since additional analyses are often required. The major advantage of EnzymeMiner over existing protein sources is the flexibility of input and concise annotation-rich interactive presentation of results. The user can input custom queries and a custom description of essential residues to focus the search on specific protein families or subfamilies. The output of EnzymeMiner is an interactive selection table containing the annotated sequences divided into sheets based on various criteria. The table helps to select a diverse set of sequences for experimental characterization. Two key prioritization criteria are (i) the predicted solubility score, which can be used to prioritize the identified sequences and increase the chance of finding enzymes with soluble protein expression, and (ii) the sequence identity to query sequences complemented with an interactive SSN displayed directly on the web, which can be used to find diverse sequences. Additionally, source organism and domain annotations help to select sequences with diverse properties. EnzymeMiner is a universal tool applicable to any enzyme family. It reduces the time needed for data gathering, multi-step analysis and sequence prioritization from days to hours. All the EnzymeMiner features are implemented directly on the web server and no external tools are required. The web server was optimized for modern browsers including Chrome, Firefox and Safari. An EnzymeMiner job can take a few hours or days to compute, depending on the current load of the server. In the next EnzymeMiner version, we plan three major improvements. First, we will implement automated tertiary structure prediction based on homology modeling and threading for all identified sequences. The structural predictions will allow subsequent analysis of active site pockets/cavities and access tunnels. Structural features will significantly enrich the set of annotations and help to identify additional attractive targets for experimental characterization. Second, we will implement automated periodical mining. When enabled, EnzymeMiner will rerun the analysis periodically and inform the user about novel sequences found since the last search. Finally, we will implement a wizard for automated selection of hits based on input criteria provided by a user.

ACKNOWLEDGEMENTS

We thank the participants of the 1st Hands-on Computational Enzyme Design Course (Brno, Czech Republic) for giving valuable feedback on the EnzymeMiner user interface. Their comments inspired us to make the sequence similarity network visualization more interactive.

FUNDING

Czech Ministry of Education [857560, 02.1.01/0.0/0.0/18_046/0015975, CZ.02.1.01/0.0/0.0/16_026/0008451, CZ.02.1.01/0.0/0.0/16_019/0000868, LQ1602]; European Commission [720776, 814418]; AI Methods for Cybersecurity and Control Systems project of the Brno University of Technology [FIT-S-20-6293]; Computational resources were supplied by the project 'e-Infrastruktura CZ' [e-INFRA LM2018140] provided within the program Projects of Large Research, Development and Innovations Infrastructures, and by the ELIXIR-CZ project [LM2015047], part of the international ELIXIR infrastructure. Funding for open access charge: Czech Ministry of Education.

Conflict of interest statement. None declared.

REFERENCES

1. Sayers,E.W., Agarwala,R., Bolton,E.E., Brister,J.R., Canese,K., Clark,K., Connor,R., Fiorini,N., Funk,K., Hefferon,T. *et al.* (2019) Database resources of the national center for biotechnology information. *Nucleic Acids Res.*, **47**, D23–D28.
2. UniProt Consortium (2019) UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Res.*, **47**, D506–D515.
3. Colin,P.-Y., Kintses,B., Gielen,F., Miton,C.M., Fischer,G., Mohamed,M.F., Hyvönen,M., Morgavi,D.P., Janssen,D.B. and Hollfelder,F. (2015) Ultrahigh-throughput discovery of promiscuous enzymes by picodroplet functional metagenomics. *Nat. Commun.*, **6**, 1–12.
4. Beneyton,T., Thomas,S., Griffiths,A.D., Nicaud,J.-M., Drevelle,A. and Rossignol,T. (2017) Droplet-based microfluidic high-throughput screening of heterologous enzymes secreted by the yeast Yarrowia lipolytica. *Microb. Cell Fact.*, **16**, 18.
5. Vanacek,P., Sebestova,E., Babkova,P., Bidmanova,S., Daniel,L., Dvorak,P., Stepankova,V., Chaloupkova,R., Brezovsky,J., Prokop,Z. *et al.* (2018) Exploration of enzyme diversity by integrating bioinformatics with expression analysis and biochemical characterization. *ACS Catal.*, **8**, 2402–2412.
6. El-Gebali,S., Mistry,J., Bateman,A., Eddy,S.R., Luciani,A., Potter,S.C., Qureshi,M., Richardson,L.J., Salazar,G.A., Smart,A. *et al.* (2019) The Pfam protein families database in 2019. *Nucleic Acids Res.*, **47**, D427–D432.
7. Li,Y., Wang,S., Umarov,R., Xie,B., Fan,M., Li,L. and Gao,X. (2018) DEEPre: sequence-based enzyme EC number prediction by deep learning. *Bioinformatics*, **34**, 760–769.
8. Zhou,N., Jiang,Y., Bergquist,T.R., Lee,A.J., Kacsoh,B.Z., Crocker,A.W., Lewis,K.A., Georghiou,G., Nguyen,H.N., Hamid,M.N. *et al.* (2019) The CAFA challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome Biol.*, **20**, 244.
9. Mak,W.S., Tran,S., Marcheschi,R., Bertolani,S., Thompson,J., Baker,D., Liao,J.C. and Siegel,J.B. (2015) Integrative genomic mining for enzyme function to enable engineering of a non-natural biosynthetic pathway. *Nat. Commun.*, **6**, 1–10.
10. Altschul,S. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.
11. Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, **26**, 2460–2461.
12. Sievers,F., Wilm,A., Dineen,D., Gibson,T.J., Karplus,K., Li,W., Lopez,R., McWilliam,H., Remmert,M., Söding,J. *et al.* (2011) Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal Omega. *Mol. Syst. Biol.*, **7**, 539.
13. Krogh,A., Larsson,B., von Heijne,G. and Sonnhammer,E.L. (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J. Mol. Biol.*, **305**, 567–580.
14. Quevillon,E., Silventoinen,V., Pillai,S., Harte,N., Mulder,N., Apweiler,R. and Lopez,R. (2005) InterProScan: protein domains identifier. *Nucleic Acids Res.*, **33**, W116–W120.
15. Federhen,S. (2012) The NCBI Taxonomy database. *Nucleic Acids Res.*, **40**, D136–D143.
16. Barrett,T., Clark,K., Gevorgyan,R., Gorelenkov,V., Gribov,E., Karsch-Mizrachi,I., Kimelman,M., Pruitt,K.D., Resenchuk,S., Tatusova,T. *et al.* (2012) BioProject and BioSample databases at NCBI: facilitating capture and organization of metadata. *Nucleic Acids Res.*, **40**, D57–D63.
17. Musil,M., Konegger,H., Hon,J., Bednar,D. and Damborsky,J. (2019) Computational design of Stable and Soluble Biocatalysts. *ACS Catal.*, **9**, 1033–1054.
18. Steinegger,M. and Söding,J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.*, **35**, 1026–1028.
19. Shannon,P. (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, **13**, 2498–2504.
20. Copp,J.N., Akiva,E., Babbitt,P.C. and Tokuriki,N. (2018) Revealing unexplored sequence-function space using sequence similarity networks. *Biochemistry*, **57**, 4651–4662.
21. Gerlt,J.A., Bouvier,J.T., Davidson,D.B., Imker,H.J., Sadkhin,B., Slater,D.R. and Whalen,K.L. (2015) Enzyme Function Initiative-Enzyme Similarity Tool (EFI-EST): A web tool for generating protein sequence similarity networks. *Biochim. Biophys. Acta (BBA) - Proteins Proteomics*, **1854**, 1019–1037.
22. Berman,H.M., Westbrook,J., Feng,Z., Gilliland,G., Bhat,T.N., Weissig,H., Shindyalov,I.N. and Bourne,P.E. (2000) The Protein Data Bank. *Nucleic Acids Res.*, **28**, 235–242.
23. Klesmith,J.R., Bacik,J.-P., Wrenbeck,E.E., Michalczyk,R. and Whitehead,T.A. (2017) Trade-offs between enzyme fitness and solubility illuminated by deep mutational scanning. *Proc. Natl Acad. Sci. U.S.A.*, **114**, 2265–2270.

SoluProt: Prediction of Soluble Protein Expression in Escherichia coli.

Bioinformatics, 37(1), 2021, 23–28
doi: 10.1093/bioinformatics/btaa1102
Advance Access Publication Date: 8 January 2021
Original Paper

Sequence analysis

SoluProt: prediction of soluble protein expression in *Escherichia coli*

Jiri Hon[1,2,3], Martin Marusiak[3], Tomas Martinek[3], Antonin Kunka[1,2], Jaroslav Zendulka[3], David Bednar[1,2,*] and Jiri Damborsky [1,2,*]

[1]Loschmidt Laboratories, Centre for Toxic Compounds in the Environment RECETOX and Department of Experimental Biology, Faculty of Science, Masaryk University, Brno 625 00, Czech Republic, [2]International Clinical Research Center, St. Anne's University Hospital Brno, Brno 656 91, Czech Republic and [3]IT4Innovations Centre of Excellence, Faculty of Information Technology, Brno University of Technology, Brno 612 66, Czech Republic

*To whom correspondence should be addressed.
Associate Editor: Jinbo Xu

Received on October 3, 2020; revised on December 5, 2020; editorial decision on December 24, 2020; accepted on December 28, 2020

Abstract

Motivation: Poor protein solubility hinders the production of many therapeutic and industrially useful proteins. Experimental efforts to increase solubility are plagued by low success rates and often reduce biological activity. Computational prediction of protein expressibility and solubility in *Escherichia coli* using only sequence information could reduce the cost of experimental studies by enabling prioritization of highly soluble proteins.
Results: A new tool for sequence-based prediction of soluble protein expression in *E.coli*, SoluProt, was created using the gradient boosting machine technique with the TargetTrack database as a training set. When evaluated against a balanced independent test set derived from the NESG database, SoluProt's accuracy of 58.5% and AUC of 0.62 exceeded those of a suite of alternative solubility prediction tools. There is also evidence that it could significantly increase the success rate of experimental protein studies. SoluProt is freely available as a standalone program and a user-friendly webserver at https://loschmidt.chemi.muni.cz/soluprot/.
Availability and implementation: https://loschmidt.chemi.muni.cz/soluprot/.
Contact: jiri@chemi.muni.cz
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Low protein solubility causes severe problems in protein science and industry; insufficient protein solubility is probably the most common cause of failure of protein production pipelines. The importance of solubility is underlined by the findings of the large-scale Protein Structure Initiative (PSI) project (Berman *et al.*, 2017), which sought to produce thousands of protein sequences from different organisms, crystallize them and resolve their tertiary structure. Unfortunately, in most cases it proved impossible to produce the target proteins in soluble form. The inherent low solubility of natural enzymes also limits the success of emerging high-throughput pipelines that explore protein databases to identify novel enzymes with diverse functions (Hon *et al.*, 2020; Vanacek *et al.*, 2018). Given the rapid growth of protein sequence databases driven by the capabilities of next-generation sequencing technologies, there is an urgent need to focus only on potentially soluble targets to avoid wasting resources on hard-to-produce orthologs. Solubility is thus a key attribute when choosing protein targets for experimental characterization (Vanacek *et al.*, 2018). Strictly speaking, solubility is a thermodynamic parameter defined as the protein's concentration in a saturated solution in equilibrium with a solid phase under specific conditions. However, it is challenging to quantitatively measure the solubility of large sets of proteins (Kramer *et al.*, 2012), so there is little quantitative experimental data on protein solubility. Moreover, this definition of solubility is too narrow to encompass many of the practical problems that may occur during protein production with common expression systems. Therefore, inspired by existing tools (Supplementary Table S1) (Agostini *et al.*, 2014; Khurana *et al.*, 2018; Raimondi *et al.*, 2020; Smialowski *et al.*, 2012), available data (Berman *et al.*, 2017) and laboratory practice, we use a slightly extended definition of protein solubility in this work. Specifically, by solubility, we mean the probability of soluble protein (over)expression in *Escherichia coli* cells. The difference from the classical thermodynamic solubility is in the perception of the insoluble class. We assume that insoluble proteins were either not expressed or were expressed in the insoluble form.

Solubility depends on many extrinsic and intrinsic factors. Extrinsic factors are dictated by the choice of expression system and the experimental conditions used in protein production. Expression systems may be either *in vivo* or *in vitro* (Carlson *et al.*, 2012; Rosano and Ceccarelli, 2014). *In vivo* protein expression is induced inside living cells of a host organism, whereas *in vitro* expression relies on the use of cell-free translational systems. Solubility can be increased by adjusting extrinsic solubility factors, especially by using different mutated host strains, codon optimization, coexpression of chaperones and foldases, lowering cultivation temperatures and adding suitable fusion partners (Costa *et al.*, 2014). However, tuning the expression system or experimental conditions is not always sufficient to confer solubility, and is not feasible in high-throughput protein production pipelines. If extrinsic factors cannot be varied, protein solubility will depend only on the intrinsic properties of the protein sequence. Unfortunately, the relationship between a protein's sequence and its solubility is poorly understood, mainly due to a lack of reproducible quantitative solubility measurements (Kramer *et al.*, 2012). Recent protein engineering studies suggest that charged amino acids on the protein surface are key intrinsic determinants of solubility (Carballo-Amador *et al.*, 2019; Chan *et al.*, 2013; Sankar *et al.*, 2018). However, this knowledge cannot be directly used for solubility prediction due to a lack of structural data. Despite the continuous growth of structural databases (Burley *et al.*, 2019), the structures of proteins of interest are generally unknown, and the limited availability of template structures prevents their accurate computational prediction.

The simultaneous effects of extrinsic and intrinsic factors make solubility prediction challenging. For example, the prediction of solubility from sequence data using machine learning is hindered by the high level of noise in typical training datasets due to the influence of diverse extrinsic variables. Because the molecular mechanisms governing protein solubility are poorly understood, recent solubility prediction tools rely heavily on statistical analysis and machine learning, using previously reported experimental data to train and validate model parameters. One of the most widely used data sources is the TargetTrack database (Berman *et al.*, 2017), formerly known as PepcDB or TargetDB, which integrates information from the Protein Structure Initiative projects. This database contains data from over 900 000 protein crystallization trials involving almost 300 000 unique protein sequences, which are referred to as targets. The database does not contain solubility data per se, but target proteins can be considered soluble if they were successfully purified in the experimental trials. A major limitation of this database is the low quality of its annotations. For example, reasons for failure are generally not provided for unsuccessful crystallization attempts. Therefore, it is impossible to distinguish failures due to insolubility from failures due to other problems later in the experimental pipeline. Second, the experimental protocols used for protein production and crystallization are described in free text with no internal structure, making it hard to automatically extract information about experimental conditions and expression systems for a given target. Filtering is therefore needed to reduce noise before using the TargetTrack data for model training. However, the application of stringent filtering rules to the target annotations can dramatically reduce the number of usable records.

eSOL is another well-known and commonly used solubility database (Niwa *et al.*, 2009, 2012) that contains experimentally measured solubilities for over 3 000 *E.coli* proteins produced in the PURE (Shimizu *et al.*, 2001) cell-free expression system. eSOL is an impressive collection of highly homogenous data but has its own limitations. First, it only contains data on proteins originating from *E.coli*. Second, it has relatively little negative data; adding the three main cytosolic *E.coli* chaperones (TF, DnaKJE and GroEL/GroES) to the PURE expression system reduced the number of insoluble proteins from 788 to 24 (Niwa *et al.*, 2012). eSOL is a valuable source of exact solubility data that were generated using a robust pipeline and provide a good quantitative measure of thermodynamic solubility. However, these data cannot be used to assess solubility according to our expanded definition, which also encompasses expressibility.

The relationship between protein sequence and solubility has been studied for over 30 years, leading to the development of several predictive models and software tools. There are 11 such models or tools that use definitions of solubility like that described above and take protein sequences as their sole input. These are the revised Wilkinson-Harrison model (rWH) (Davis *et al.*, 1999; Wilkinson and Harrison, 1991), SOLpro (Magnan *et al.*, 2009), RPSP (Diaz *et al.*, 2010), PROSO II (Smialowski *et al.*, 2012), ccSOL omics (Agostini *et al.*, 2012, 2014), ESPRESSO (Hirose and Noguchi, 2013), CamSol (Sormanni *et al.*, 2015), Protein-Sol (Hebditch *et al.*, 2017), DeepSol (Khurana *et al.*, 2018), SKADE (Raimondi *et al.*, 2020) and the Solubility-weighted index (SWI) (Bhandari *et al.*, 2020). However, the accuracy of these tools is limited, and there is clear room for improvement. Additionally, these tools exhibit poor generality when used to make predictions based on previously unseen data. A comprehensive review of advances in solubility prediction, including predictors that use protein structures as inputs, was published recently (Musil *et al.*, 2019). Here, we present a novel machine learning based tool, SoluProt, for predicting soluble expression from protein sequence data. SoluProt benefits from thorough dataset pre-processing and predicts soluble expression more accurately than previously reported methods.

2 SoluProt training and test set

We used the TargetTrack database to build the *SoluProt training set*. Since this database does not directly provide solubility information, we inferred solubility computationally, using an approach similar to those adopted previously (Magnan *et al.*, 2009; Smialowski *et al.*, 2012). A protein was considered *soluble* if it was recorded as having reached a soluble experimental state or any subsequent state requiring soluble expression (Supplementary Table S2). If failed expression or purification was mentioned in the experiment record's stop status, the protein was labeled *insoluble*. In contrast to a previous approach (Smialowski *et al.*, 2012), we required an explicit stop status relating to insolubility to reduce the frequency of incorrect classification of insoluble sequences. To improve the quality of the training set, we also performed several additional steps to clean the data.

Most importantly, we performed keyword matching combined with manual checking of TargetTrack annotations to extract only proteins expressed in the most common host organism, *E.coli*. This was necessary because a protein soluble in one organism might be insoluble in another. By focusing solely on the most common expression system, we reduced the noise in the training data. We also used specific keywords to search the unstructured descriptions of experimental protocols provided in the TargetTrack database (Supplementary Table S3). Generic search phrases like '*E.coli*' or '*Escherichia coli*' were used to identify potential *E.coli* related protocols. These protocols were then manually checked and confirmed (Supplementary Table S4). A full list of 248 TargetTrack protocols signifying expression in *E.coli* is available at the SoluProt website.

We next identified transmembrane proteins in the dataset based on direct annotations from the TargetTrack database and predictions generated using TOPCONS (Tsirigos *et al.*, 2015) with default settings. The transmembrane proteins were then removed, along with sequences shorter than 20 amino acids, and sequences with undefined residues. We also removed sequences that had been classified as insoluble but for which a protein structure was available in the Protein Data Bank (PDB) (Berman, 2000). To this end, we compiled an *E.coli* PDB subset containing sequences of proteins whose structures had been solved by NMR or X-ray crystallography and which had been expressed in *E.coli* according to the PDB annotations (64 416 sequences, downloaded April 4, 2018). Because both NMR and X-ray crystallography require soluble proteins, any protein in this PDB subset can be considered soluble in *E.coli*. This step reflects advances in molecular biology: methodological developments have made it possible to produce and crystallize some proteins that were previously considered insoluble.

Finally, we reduced the sequence redundancy in the training set by clustering to 25% identity using MMseqs2 (Steinegger and

Söding, 2017) and retaining only representative sequences from each cluster. This was done separately for positive and negative samples to avoid simplifying the prediction problem. We balanced the number of soluble and insoluble samples such that both classes were equally represented. Additionally, we balanced the sequence length distribution so that length alone would not play a dominant role in the predictions. Sequence length correlates with protein solubility—larger proteins are usually less soluble. However, we wanted to suppress its influence in the model because we anticipate that SoluProt would mainly be used to prioritize proteins of similar lengths, usually from a single protein family. A typical expected use case is that of the EnzymeMiner web server (Hon *et al.*, 2020) for automated mining of soluble enzymes. A prediction model relying heavily on sequence length would not perform well in this use case.

The *SoluProt test set* was built from a dataset generated by the North East Structural Consortium (NESG), which represents 9644 proteins expressed in *E.coli* using a unified production pipeline (Price *et al.*, 2011). The dataset contains two integer scores ranging from 0 to 5 for each target, indicating the protein's level of expression and the soluble fraction recovery. The reproducibility of the experimental results in the dataset was validated by performing repeat measurements for selected targets. The NESG dataset targets are included in the TargetTrack database because the NESG participated in the PSI project. However, the expression and solubility levels from the NESG dataset were not included in the TargetTrack database; instead, they were provided to us directly by the authors of the original study (W. Nicholson Price II, personal communication). The high consistency and quality of the NESG dataset make it suitable for benchmarking purposes. We processed the NESG dataset using the same procedure as the training set, although the computational solubility derivation and expression system filtration steps were omitted because they were pointless in this case. Instead, we transformed the solubility levels into binary classes: all proteins with a solubility level of 1 or above were considered soluble and all others insoluble.

Finally, we ensured that no pair consisting of a sequence from the test set and a sequence from the training set had a global sequence identity above 25% as calculated using the USEARCH software (Edgar, 2010). This made the test set more independent because it ensured that predictions were not validated against data similar to those used during training. In total, 11 436 protein sequences remained in the *SoluProt training set* and 3 100 in the independent *SoluProt test set*. Both datasets had equal numbers of soluble and insoluble samples with balanced sequence length distributions (Supplementary Fig. S1). The datasets are available at the SoluProt website. The dataset construction steps are summarized in Supplementary Table S5.

3 Prediction model

The SoluProt predictor is implemented in Python using scikit-learn (Pedregosa *et al.*, 2011), Biopython (Cock *et al.*, 2009) and pandas (McKinney, 2010) libraries. We used a gradient boosting machine (GBM) (Friedman, 2001) to generate the predictive model. Prediction features were selected from a set of 251 sequence characteristics that were divided into eight groups: (i) single amino acid content (20 features), (ii) amino acid dimer content (210 features), (iii), sequence physicochemical features (12 features, Supplementary Table S6), (iv) average flexibility as computed by DynaMine (Cilia *et al.*, 2014) (1 feature), (v) secondary structure content as predicted by FESS (Piovesan *et al.*, 2017) (3 features), (vi) average disorder as predicted by ESPRITZ (Walsh *et al.*, 2012) (1 feature), (vii) content of amino acids in transmembrane helices as predicted by TMHMM (Krogh *et al.*, 2001) (3 features) and (viii) maximum identity to the *E.coli* PDB subset as calculated using USEARCH (1 feature). All sequences equal to any sequence from the test set were excluded from the *E.coli* PDB subset for the calculation of maximum identity. The objective was to eliminate even the indirect presence of test set sequences from model training. We standardized all features by subtracting the mean and scaling to unit variance. The means and variances were calculated using the training set.

We removed correlated features in two steps. First, we fitted a GBM with default parameters using the full training set and all features. Second, we calculated Pearson's correlation coefficient for each pair of features. If the correlation between any two features exceeded 0.75, we removed the feature with the lesser importance in the fitted GBM model. We also removed irrelevant features using LASSO (Tibshirani, 1996). LASSO's alpha parameter was optimized to maximize the mean AUC of the GBM model with default parameters over 5-fold cross-validation. The alpha parameter was varied between 0.08 to 0 with a step size of 6.25×10^{-4}; its optimal value was 0.005. In total, 96 features were selected for inclusion in the predictive model (Supplementary Table S7). The DynaMine, FESS and ESPRITZ features were not included in the final feature set.

We next optimized the hyperparameters of the GBM model, using an iterative 7-stage strategy to maximize the mean AUC over 5-fold cross-validation using the training set (Supplementary Table S8). In each stage, one or two parameters were optimized using grid search; other parameters were left either at their final values from the previous stages or at the default value if the parameter had not yet been optimized. The best GBM model achieved mean AUC values of 0.85 ± 0.003 for the training part and 0.72 ± 0.02 for the validation part. Overall, the feature selection and hyperparameter optimization had little effect on the mean AUC: without these measures, the mean AUC values for the training and validation sets were 0.83 ± 0.003 and 0.72 ± 0.02, respectively. The main benefit of the feature selection and parameter tuning steps was that they reduced the number of features and thus made the feature calculation step roughly two times faster.

Finally, we used the best GBM hyperparameters to train the final SoluProt model using the full training set. The resulting model had an AUC of 0.84 and an accuracy of 76% for the full training set. The five most important features according to the GBM are: (i) maximum identity to the *E.coli* PDB subset (14.5%), (ii) isoelectric point (6.2%), (iii) predicted number of amino acids in transmembrane helices in the first sixty amino acids of the protein (4.2%), (iv) lysine content (4.0%) and (v) glutamine content (3.5%) (Supplementary Table S7).

4 Performance evaluation and comparison

We used the SoluProt test set to evaluate and compare SoluProt to 11 previously published tools. The evaluation relied on both threshold-independent (area under the ROC curve) and threshold-dependent metrics (accuracy, Matthew's correlation coefficient and confusion matrices). For the threshold-dependent metrics, we applied a threshold of 0.5 or the thresholds recommended by the authors of the corresponding method (Table 1). SoluProt achieved the highest accuracy (58.5%) and the greatest AUC (0.62) of the

Table 1. Performance of various solubility predictors using the balanced SoluProt test set of 3100 sequences

Method	AUC	T	ACC	MCC	TP	TN	FP	FN
SoluProt	0.62	0.50	58.5%	0.17	939	873	677	611
PROSO II	0.60	0.60	58.0%	0.17	630	1167	383	920
SWI	0.60	0.50	55.9%	0.13	1206	527	1023	344
CamSol	0.57	1.00	54.1%	0.08	676	1001	549	874
ESPRESSO	0.56	0.50	53.8%	0.08	1003	664	886	547
rWH	0.55	0.50	54.0%	0.08	670	1005	545	880
DeepSol	0.55	0.50	52.9%	0.09	230	1409	141	1320
Protein-Sol	0.54	0.45	51.6%	0.03	1056	544	1006	494
SOLpro	0.53	0.50	52.0%	0.04	654	959	591	896
SKADE	0.51	0.50	49.2%	–0.03	159	1366	184	1391
ccSOL omics	0.51	0.50	50.8%	0.02	884	690	860	666
RPSP	0.50	0.50	49.8%	0.00	501	1044	506	1049

Note: The different definitions of solubility and target expression system (Supplementary Table S1) should be considered when comparing the performance of individual tools.

AUC—area under the ROC curve, T—threshold for the soluble class, ACC—accuracy, MCC—Matthew's correlation coefficient, TP—true positives, TN—true negatives, FP—false positives, FN—false negatives.

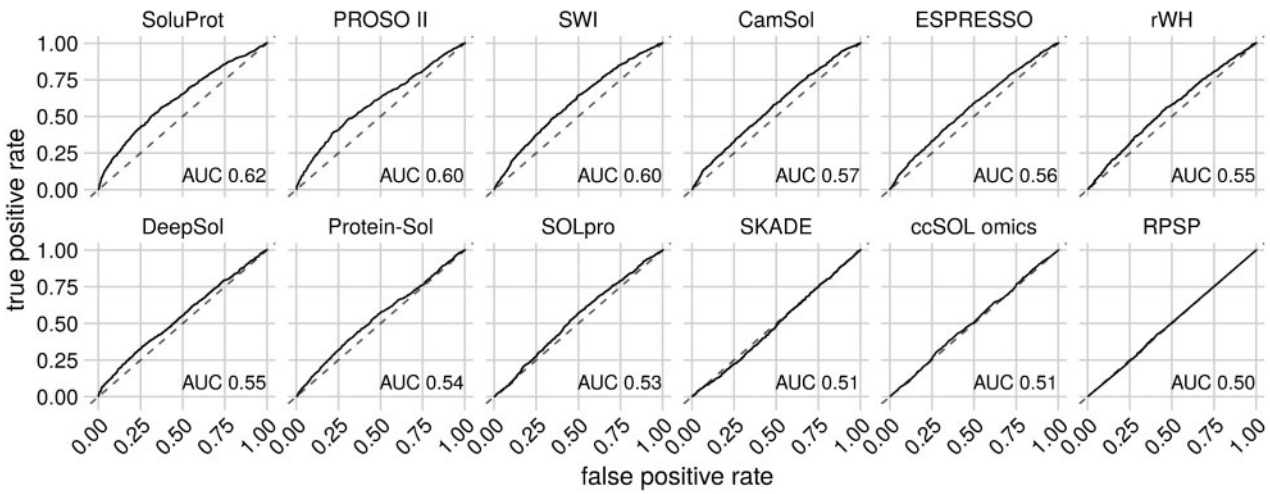

Fig. 1. Receiver operating curves (ROC) calculated for the balanced SoluProt test set of 3100 sequences. The predictors are ordered by the area under the receiver operating curve (AUC)

Table 2. Overlaps between the SoluProt test set and available training sets

Dataset	Size	Test set overlap	TP	TN	FP	FN
PROSO II initial	129643	2952 (95.2%)	951	1437	50	514
DeepSol/SKADE	69420	2294 (74.0%)	737	1130	67	360
SWI	12216	820 (26.5%)	537	210	53	20
SOLpro	17408	480 (15.5%)	178	120	39	143

Note: Two sequences were considered identical if their global sequence identity reported by USEARCH was 100%. Differences in solubility annotations for identical sequences were quantified using confusion matrix terms (TP, TN, FP and FN). The solubility annotations of the SoluProt test set are assumed to reflect the true solubilities of the proteins.

TP—true positives, TN—true negatives, FP—false positives, FN—false negatives. [a] DeepSol and SKADE share the same training set.

tested tools when evaluated against the SoluProt test set (Table 1 and Fig. 1),followed by PROSO II and SWI.

While the SoluProt test set is independent of the SoluProt training set, other tools' training sets might overlap with our test set. Therefore, we compared the SoluProt test set to the training sets of DeepSol, SKADE, SWI and SOLpro to quantify their overlaps (Table 2). DeepSol and SKADE have a common training set, which showed the largest overlap (74.0%), followed by the SWI training set (26.5%) and the SOLpro training set (15.5%). SWI benefits from the overlap; it was the third-best tool in our comparison. DeepSol and SKADE ranked 7th and 12th by accuracy with respect to the SoluProt test set despite having the greatest proportion of test sequences in their training set. This comparatively poor performance can be partly explained by differences in solubility annotations between the DeepSol training set and the SoluProt test set (Table 2): 360 (11.6% of the total) sequences annotated as insoluble in the DeepSol training set were annotated as soluble in the SoluProt test set. The total number of disagreements (the sum of false positives and false negatives) ranged from 336 to 551, depending on the binarization threshold applied to the SoluProt test set (Supplementary Table S9). No training set was published for PROSO II; only an initial set of soluble and insoluble sequences without pre-processing is available. However, the initial set exhibits 95.2% overlap with the SoluProt test set. Therefore, we expect the overlap of the PROSO II training set to also be very high, like the DeepSol training set. Unfortunately, the training sets of other previously developed tools have not been published, preventing a more comprehensive comparison.

The absolute accuracy of the available solubility prediction tools is low (below 60%), so there is clearly room for improvement. Nevertheless, SoluProt and other tools can be useful for protein sequence prioritization (Fig. 2), i.e. for selecting a small number of sequences for in-depth experimental characterization from a large database of several hundreds or thousands of sequences. Specifically, predicted solubility values can be used to select a limited number of high-scoring protein sequences. For example, if we use SoluProt predictions to order the SoluProt test set and remove all sequences bar the 10% with the highest scores, we get 232 true positives, i.e. 49.7% more true positives than would be expected with blind selection (155 true positives). This shows that despite their limited accuracy, current solubility predictors are valuable for protein sequence prioritization and can increase the success rate of experimental protein studies.

5 Conclusions

We have developed a novel method and software tool, SoluProt, for sequence-based prediction of soluble protein expression in *E.coli*. The tool simultaneously predicts the solubility and expressibility of the proteins under consideration. SoluProt achieved a higher accuracy (58.5%) and AUC (0.62) than a suite of alternative solubility prediction tools when evaluated using the balanced independent SoluProt test set of 3100 sequences. PROSO II, SWI and CamSol were the next best tools, achieving accuracies of 58.0%, 55.9% and 54.1%, respectively. SoluProt also performed well in protein prioritization. The main strengths of SoluProt are that it was trained using a dataset generated by thorough pre-processing of the noisy TargetTrack data, and was validated using a high-quality independent test set.

Surprisingly, the recently reported DeepSol (Khurana *et al.*, 2018) and SKADE (Raimondi *et al.*, 2020) tools, which are based on deep learning methods, performed worse than the simpler and mostly older methods PROSO II (Smialowski *et al.*, 2012), SWI (Bhandari *et al.*, 2020) and CamSol (Sormanni *et al.*, 2015) in our comparison. This may be partly due to the overlap of their training set with our test set and disagreements between these sets with respect to the solubility of certain sequences.

The SoluProt predictor is available via a user-friendly web server or as a standalone software package at https://loschmidt.chemi.muni.cz/soluprot/. The SoluProt web server has already predicted the solubility of over 4700 unique protein sequences in ten months since its launch in February 2020. It has also been integrated into the web server EnzymeMiner (Hon *et al.*, 2020) for automated

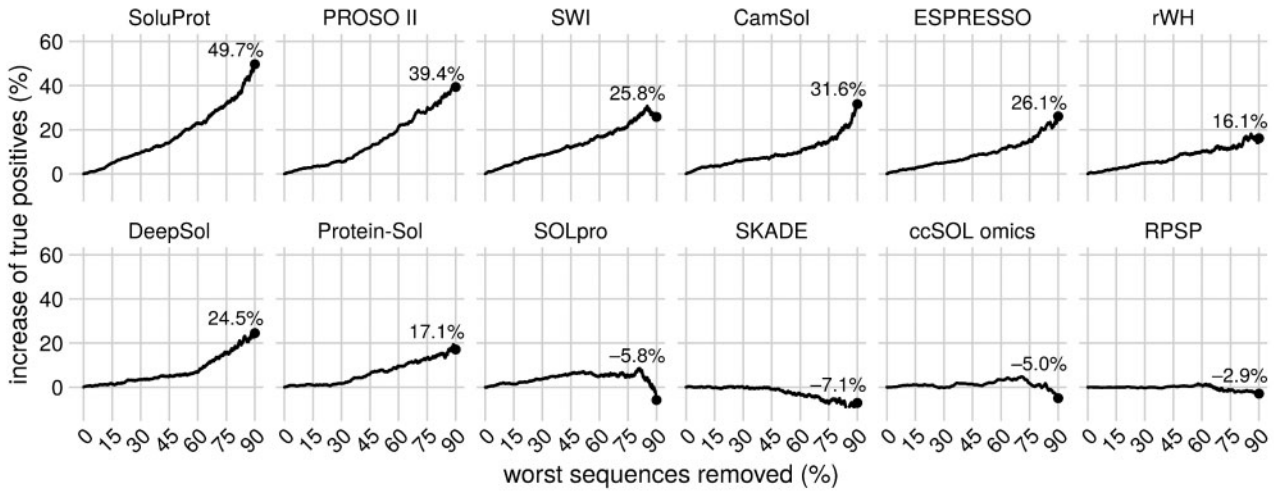

Fig. 2. Increases in the number of true positives resulting from sequence prioritization using the tested solubility prediction tools. The SoluProt test set sequences were ordered by predicted solubility based on each predictor's output, and a variable percentage of the sequences with the worst predicted solubility was then removed. The increase in the number of true positives was then calculated relative to a baseline random selection. For example, upon randomly removing 90% of the test set sequences (2790 samples), we would expect half of the remaining 310 sequences to be true positives

mining of novel soluble enzymes from protein databases (https://loschmidt.chemi.muni.cz/enzymeminer/).

Funding

This work was supported by Czech Ministry of Education [857560, 02.1.01/0.0/0.0/18_046/0015975, CZ.02.1.01/0.0/0.0/16_026/0008451, LQ1602]; Czech Grant Agency (20-15915Y); European Commission [857560, 720776, 814418]; and AI Methods for Cybersecurity and Control Systems project of the Brno University of Technology [FIT-S-20-6293]. Computational resources were supplied by the project 'e-Infrastruktura CZ' [e-INFRA LM2018140] and by the ELIXIR-CZ [LM2018131]. Funding for open access charge: Czech Ministry of Education.

Conflict of Interest: none declared.

References

Agostini,F. *et al.* (2014) ccSOL omics: a webserver for solubility prediction of endogenous and heterologous expression in *Escherichia coli*. *Bioinformatics*, **30**, 2975–2977.

Agostini,F. *et al.* (2012) Sequence-based prediction of protein solubility. *J. Mol. Biol.*, **421**, 237–241.

Berman,H.M. *et al.* (2017) Protein Structure Initiative – TargetTrack 2000-2017 – all data files. *Zenodo*. doi:10.5281/zenodo.821654.

Berman,H.M. (2000) The Protein Data Bank. *Nucleic Acids Res.*, **28**, 235–242.

Bhandari,B.K. *et al.* (2020) Solubility-Weighted Index: fast and accurate prediction of protein solubility. *Bioinformatics*, **36**, 4691–4698.

Burley,S.K. *et al.* (2019) RCSB Protein Data Bank: biological macromolecular structures enabling research and education in fundamental biology, biomedicine, biotechnology and energy. *Nucleic Acids Res.*, **47**, D464.

Carballo-Amador,M.A. *et al.* (2019) Surface patches on recombinant erythropoietin predict protein solubility: engineering proteins to minimise aggregation. *BMC Biotechnology*, **19**, 26.

Carlson,E.D. *et al.* (2012) Cell-free protein synthesis: applications come of age. *Biotechnol. Adv.*, **30**, 1185–1194.

Chan,P. *et al.* (2013) Soluble expression of proteins correlates with a lack of positively-charged surface. *Sci. Rep.*, **3**, 3333.

Cilia,E. *et al.* (2014) The DynaMine webserver: predicting protein dynamics from sequence. *Nucleic Acids Res.*, **42**, W264–W270.

Cock,P.J.A. *et al.* (2009) Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics*, **25**, 1422–1423.

Costa,S. *et al.* (2014) Fusion tags for protein solubility, purification and immunogenicity in *Escherichia coli*: the novel Fh8 system. *Front. Microbiol.*, **5**, 63.

Davis,G.D. *et al.* (1999) New fusion protein systems designed to give soluble expression in Escherichia coli. *Biotechnol. Bioeng.*, **65**, 382–388.

Diaz,A.A. *et al.* (2010) Prediction of protein solubility in *Escherichia coli* using logistic regression. *Biotechnol. Bioeng.*, **105**, 374–383.

Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, **26**, 2460–2461.

Friedman,J.H. (2001) Greedy function approximation: a gradient boosting machine. *Ann. Stat.*, **29**, 1189–1232.

Hebditch,M. *et al.* (2017) Protein–Sol: a web tool for predicting protein solubility from sequence. *Bioinformatics*, **33**, 3098–3100.

Hirose,S. and Noguchi,T. (2013) ESPRESSO: a system for estimating protein expression and solubility in protein expression systems. *Proteomics*, **13**, 1444–1456.

Hon,J. *et al.* (2020) EnzymeMiner: automated mining of soluble enzymes with diverse structures, catalytic properties and stabilities. *Nucleic Acids Res.*, **48**, W104–W109.

Khurana,S. *et al.* (2018) DeepSol: a deep learning framework for sequence-based protein solubility prediction. *Bioinformatics*, **34**, 2605–2613.

Kramer,R.M. *et al.* (2012) Toward a molecular understanding of protein solubility: increased negative surface charge correlates with increased solubility. *Biophys. J.*, **102**, 1907–1915.

Krogh,A. *et al.* (2001) Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J. Mol. Biol.*, **305**, 567–580.

Magnan,C.N. *et al.* (2009) SOLpro: accurate sequence-based prediction of protein solubility. *Bioinformatics*, **25**, 2200–2207.

McKinney,W. (2010) Data Structures for Statistical Computing in Python. In: *Proceedings of the 9th Python in Science Conference*. SciPy Organizers, Austin, Texas, pp. 56–61.

Musil,M. *et al.* (2019) Computational design of stable and soluble biocatalysts. *ACS Catal.*, **9**, 1033–1054.

Niwa,T. *et al.* (2009) Bimodal protein solubility distribution revealed by an aggregation analysis of the entire ensemble of *Escherichia coli* proteins. *Proc. Natl. Acad. Sci. USA*, **106**, 4201–4206.

Niwa,T. *et al.* (2012) Global analysis of chaperone effects using a reconstituted cell-free translation system. *Proc. Natl. Acad. Sci. USA*, **109**, 8937–8942.

Pedregosa,F. *et al.* (2011) Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.*, **12**, 2825–2830.

Piovesan,D. *et al.* (2017) FELLS: fast estimator of latent local structure. *Bioinformatics*, **33**, 1889–1891.

Price,W.N. *et al.* (2011) Large-scale experimental studies show unexpected amino acid effects on protein expression and solubility in vivo in *E. coli*. *Microb. Inf. Exp.*, **1**, 6.

Raimondi,D. *et al.* (2020) Insight into the protein solubility driving forces with neural attention. *PLoS Comput. Biol.*, **16**, e1007722.

Rosano,G.L. and Ceccarelli,E.A. (2014) Recombinant protein expression in *Escherichia coli*: advances and challenges. *Front. Microbiol.*, **5**, 172.

Sankar,K. *et al.* (2018) AggScore: prediction of aggregation-prone regions in proteins based on the distribution of surface patches. *Proteins*, **86**, 1147–1156.

Shimizu,Y. *et al.* (2001) Cell-free translation reconstituted with purified components. *Nat. Biotechnol.*, **19**, 751–755.

Smialowski,P. *et al.* (2012) PROSO II - a new method for protein solubility prediction. *FEBS J.*, **279**, 2192–2200.

Sormanni,P. *et al.* (2015) The CamSol method of rational design of protein mutants with enhanced solubility. *J. Mol. Biol.*, **427**, 478–490.

Steinegger,M. and Söding,J. (2017) MMseqs2 enables sensitive protein sequence searching for the analysis of massive data sets. *Nat. Biotechnol.*, **35**, 1026–1028.

Tibshirani,R. (1996) Regression shrinkage and selection via the Lasso. *J. R. Stat. Soc. Ser. B (Methodological)*, **58**, 267–288.

Tsirigos,K.D. *et al.* (2015) The TOPCONS web server for consensus prediction of membrane protein topology and signal peptides. *Nucleic Acids Res.*, **43**, W401–W407.

Vanacek,P. *et al.* (2018) Exploration of enzyme diversity by integrating bioinformatics with expression analysis and biochemical characterization. *ACS Catal.*, **8**, 2402–2412.

Walsh,I. *et al.* (2012) ESpritz: accurate and fast prediction of protein disorder. *Bioinformatics*, **28**, 503–509.

Wilkinson,D.L. and Harrison,R.G. (1991) Predicting the solubility of recombinant proteins in *Escherichia coli*. *Biotechnology (N.Y.)*, **9**, 443–448

FireProt: Web Server for Automated Design

of Thermostable Proteins.

Published online 26 April 2017 *Nucleic Acids Research, 2017, Vol. 45, Web Server issue* ***W393–W399***
doi: 10.1093/nar/gkx285

FireProt: web server for automated design of thermostable proteins

Milos Musil[1,2,3,†], Jan Stourac[1,3,†], Jaroslav Bendl[1,2,3], Jan Brezovsky[1,3], Zbynek Prokop[1,3], Jaroslav Zendulka[2,4], Tomas Martinek[1,2,4], David Bednar[1,3,*] and Jiri Damborsky[1,3,*]

[1]Loschmidt Laboratories, Department of Experimental Biology, Masaryk University, Brno, Czech Republic, [2]Department of Information Systems, Faculty of Information Technology, Brno University of Technology, Brno, Czech Republic, [3]International Centre for Clinical Research, St. Anne's University Hospital Brno, Brno, Czech Republic and [4]Centre of Excellence IT4Innovations, Technical University Ostrava, Ostrava

Received February 11, 2017; Revised April 02, 2017; Editorial Decision April 10, 2017; Accepted April 11, 2017

ABSTRACT

There is a continuous interest in increasing proteins stability to enhance their usability in numerous biomedical and biotechnological applications. A number of *in silico* tools for the prediction of the effect of mutations on protein stability have been developed recently. However, only single-point mutations with a small effect on protein stability are typically predicted with the existing tools and have to be followed by laborious protein expression, purification, and characterization. Here, we present FireProt, a web server for the automated design of multiple-point thermostable mutant proteins that combines structural and evolutionary information in its calculation core. FireProt utilizes sixteen tools and three protein engineering strategies for making reliable protein designs. The server is complemented with interactive, easy-to-use interface that allows users to directly analyze and optionally modify designed thermostable mutants. FireProt is freely available at http://loschmidt.chemi.muni.cz/fireprot.

INTRODUCTION

Proteins are widely used in numerous biomedical and biotechnological applications. However, naturally occurring proteins cannot usually withstand the harsh industrial environment, since they are mostly evolved to function at mild conditions (1). Protein engineering has revolutionized the utilization of naturally available proteins for different industrial applications by improving various protein features such as stability, activity or enantioselectivity to surpass their natural limitations. Protein stability is generally strongly correlated with its expression yield (2), half-life (3), serum survival time (4) and performance in the presence of denaturing agents (5). Thus, stability is one of the key determinants of proteins applicability in biotechnological processes.

In the ideal case, the saturation mutagenesis would be applied to evaluate every possible mutation on every position of the engineered protein (6). However, such a search space would be enormous and the experimental evaluation can delay the design of truly thermostable protein for months or even years. Therefore, there are demands for effective and precise predictive computation of protein stability. To satisfy this goal a number of *in silico* tools have been developed recently. Some of these tools such as EASE-MM (7), I-Mutant (8) or mCSM (9) are based on machine learning techniques. Others are using so-called energetic functions. These programs can be further categorized into two groups. The first group utilizes a physical effective energy function for simulating the fundamental forces between atoms and is represented by the programs like Rosetta (10) and Eris (11). The second group is based on statistical potentials for which the energies are derived from frequencies of residues or atom contacts reported in the datasets of experimentally characterized protein mutants, e.g. PoPMuSiC (12) and FoldX (13). However, due to the potentially antagonistic effect of mutations, only single-point mutations are usually predicted *in silico* and have to be followed by laborious and costly protein expression, purification and characterization. Single-point mutations typically enhance the melting temperature of target proteins by units of degree (3,14). A much higher degree of stabilization can be achieved by constructing multiple-point mutants (15). We have recently developed the FireProt (16), combining energy- and evolution-based approaches for reliable design of stable multiple-point mutants. The protocol includes several preceding filters that accelerate the calculation by omitting potentially deleterious mutations. FireProt is currently

*To whom correspondence should be addressed. Tel: +420 5 4949 3467; Fax: +420 5 4949 6302; Email: jiri@chemi.muni.cz; Website address: http://loschmidt.chemi.muni.cz/fireprot (type of web server: computational workflow)
Correspondence may also be addressed to David Bednar. Email: 222755@mail.muni.cz
†These authors contributed equally to the work as first authors.

available only in a stand-alone format and requires extensive experience in bioinformatics to carry out all necessary steps of the work flow. Currently, we are aware of only one server for design of stable multiple-point mutants - PROSS (17), utilizing Rosetta modeling and phylogenetic sequence information in its computation core.

Here, we present a web version of FireProt for the automated design of thermostable proteins. FireProt integrates sixteen computational tools and utilizes both sequence and structural information. FireProt web server provides users with thermostable proteins, constructed by three distinct strategies: (i) evolution-based approach, utilizing back-to-consensus analysis; (ii) energy-based approach, evaluating change in free energy upon mutation and (iii) combination of both evolution-based and energy-based approaches. In our view, it is very important to have this integrated approach, since phylogenetic analysis enables identification of the mutations stabilized by entropy, which cannot be predicted by force field calculations (Beerens *et al.*, under review). The server allows users to include preferred mutations into the thermostable protein, to generate corresponding structures and sequences for gene syntheses. Compared to the previously published FireProt protocol (16), minimum effort and no bioinformatics knowledge is required from users to calculate and analyze the results. Furthermore, all input parameters and computational protocols were optimized to minimize otherwise highly time demanding procedure. The server was complemented with a graphical interface allowing users to directly analyze the protein of interest and design multiple-point mutants.

MATERIALS AND METHODS

The basic workflow of FireProt strategy is outlined in Figure 1. In order to design a highly reliable thermostable multiple-point mutant, a protein defined by the user is annotated using several prediction tools and databases (Phase 1). With this knowledge in hand, energy- and evolution-based approach is applied to assemble a list of potentially stabilizing single-point mutations (Phase 2). Finally, three multiple-point mutants are generated in an additive manner, while removing potentially antagonistic effects of mutations (Phase 3).

Phase 1: Annotation of the protein

Initially, the user is requested to specify the protein structure, either by providing its PDB ID or by uploading a user-defined PDB file. The biological assembly of the target protein is then automatically generated by the MakeMultimer tool (http://watcut.uwaterloo.ca/tools/makemultimer/). Sequence homologs are obtained by performing a BLAST search (18) against the UniRef90 database (19), using the target protein sequence as an input query. Identified homologs are then aligned with the query protein using USEARCH (20), while sequences whose identity with the query is below or above the user defined thresholds (default: 30 and 90%) are excluded from the list of homologs. The remaining sequences are clustered using UCLUST (20), with a 90% identity threshold to remove close homologs. The cluster representatives are sorted based on the BLAST query coverage and by default, the first 200 of them are used to create a multiple sequence alignment with Clustal Omega tool (21). The multiple sequence alignment is used to: (i) estimate the conservation coefficient of each residue position in the protein based on the Jensen–Shannon entropy (22); (ii) identify correlated positions employing a consensual decision of the OMES (23), MI (24), aMIc (25), DCA (26), SCA (27), ELSC (28), McBASC (29) and (iii) analyze amino acid frequencies at individual positions within the protein.

Phase 2: Prediction of single-point mutations

In accordance with the original FireProt protocol, potentially stabilizing single-point mutations are identified via two separate branches: one relying on the estimation of the change of free energy upon mutation and second utilizing back-to-consensus approach.

The first, energy-based approach is employing FoldX and Rosetta tools that performed best on our testing dataset. Preceding filters accelerate the calculation by omitting potentially deleterious mutations. Prior to the identification of the single-point mutations itself, the target protein structure is amended and minimized. FoldX protocol is utilized to fill in the missing atoms in the residues and patched structure is consequently minimized with Rosetta minimization module. Conserved and correlated positions are immediately excluded from further analysis. It was observed that functional and structural constraints in proteins generally lead to the conservation of amino acid residues (30–33). Similarly, correlated residues ordinarily help to maintain protein function, folding or stability (34–36). Mutations conducted on these positions are therefore considered unsafe by current FireProt strategy, even though there is certainly a space for more sophisticated treatment of correlated positions, which will be further developed in future versions of FireProt server.

The remaining positions are subjected to saturation mutagenesis by using FoldX tool. Mutations with predicted ddG over given threshold (default: –1 kcal/mol) are steered away and rest is forwarded to Rosetta calculations. Finally, the mutations predicted by Rosetta as strongly stabilizing (default cut-off: –1 kcal/mol) are tagged as potential candidates for the design of the multiple-point mutants.

A high time demands of Rosetta analysis were one of the most excruciating issues with the original FireProt protocol. Even with the application of filters over 100 mutations was usually left for precise, but slow, Rosetta calculations. For this reason, we have evaluated several force fields and Rosetta protocols with the newly assembled dataset containing 1573 mutations from ProTherm database (37) and HotMuSiC dataset (38). Based on the results of the evaluations, the best trade-off between the time requirements and precision was selected. With Rosetta protocol 3, we have achieved more than tenfold increase in calculation speed while preserving high prediction accuracy. Details on dataset construction and protocols evaluation can be found in the Supplement 1 (Supplementary Tables S1–S5).

The second approach is based on the information obtained from multiple sequence alignment. The most common amino acid in each position of protein sequence often

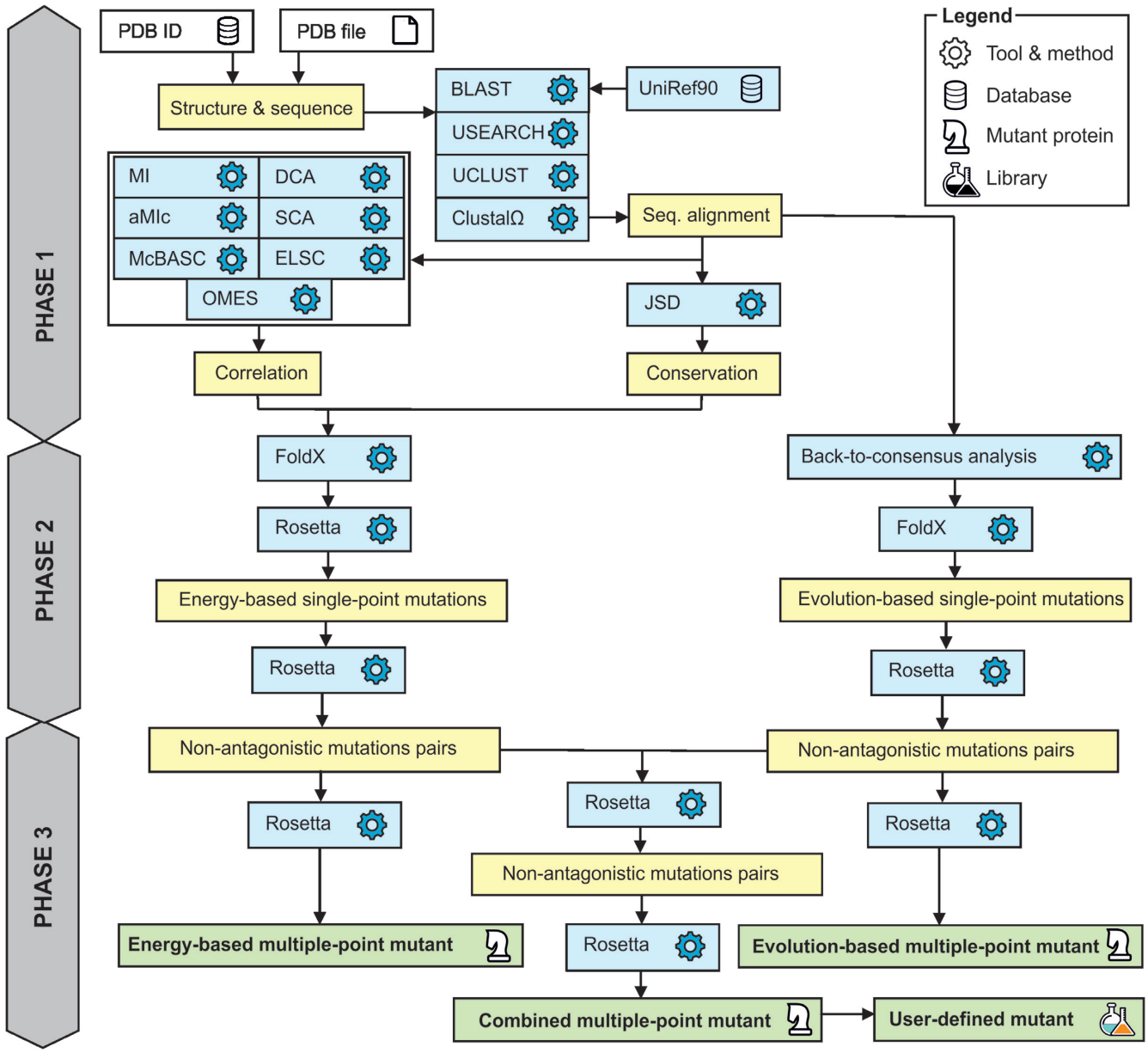

Figure 1. Workflow of FireProt strategy.

provides a non-negligible effect on protein stability (39–42). Therefore, FireProt implements majority and frequency ratio approach to identify mutations at positions where the wild-type amino acid differs from the most prevalent one. By default, the single out mutations are located in the positions where the consensus residue is present in at least 50% of all analyzed sequences (majority method) or where consensus residue frequency is 40% and is at least five times more frequent than the wild-type amino acid (frequency ratio method). These thresholds were chosen in accordance to the previously published HotSpot Wizard method (43). Selected mutations are evaluated by FoldX and the stabilizing ones are listed as candidate mutations for the engineering of multiple-point mutant.

Phase 3: Design of thermostable protein

In total, three protein designs are provided by FireProt strategy. The first design includes only the mutations from energy-based approach, the second contains the mutations suggested by the evolution-based approach and the third is the combination of both. Naturally, because of potentially antagonistic effects between individual mutations, we cannot combine individual mutations blindly.

To avoid possible clashes, FireProt strategy is trying to minimize antagonistic effects by utilizing Rosetta. In the first step, all pairs of single-point mutations within the range of 10 Å are evaluated separately for energy- and evolution-based approach. Once change in free energy is obtained for all residue pairs, FireProt starts to introduce them into the multiple-point mutant in the order based on their predicted

stability, excluding the mutations that are colliding with already included mutations. Algorithm stops once there are no mutations left or the stabilizing effect of analyzed pair drops below defined threshold.

Upon the completion of previous step, procedure is repeated this time considering only the pairs between the mutations chosen for the construction of energy- and evolution-based mutants. Finally, structures of all three mutants are modeled using the Rosetta protocol 16.

DESCRIPTION OF THE WEB SERVER

Input

The only required input to the web server is a tertiary structure of the protein of interest, provided either as a PDB ID or a user-defined PDB file. The user can then choose a predefined biological unit generated by the MakeMultimer tool or manually select chains for which the calculation should be performed. The calculations can be configured in either basic or advanced mode.

In the basic mode, user is allowed to change the setting of BLAST search and alignment construction. The advanced mode expands the list of modifiable parameters by the ones connected with: (i) the identification of consensus residues by majority and frequency ratio approach, (ii) the thresholds used by FoldX and Rosetta prediction tools and (iii) the decision threshold employed in the consensual analysis of correlated positions. Advanced mode allows expert users to fine-tune the parameters of calculation according to studied systems. However, the presented default values are optimized to provide reliable results for most of the systems and we therefore do not advice their change in the general scenarios.

Output

Upon submission, a unique identifier is assigned to each job to track the calculation and the 'Results browser' informs the user about the status of the individual steps in the FireProt workflow (Figure 2B). Once the job is finished, users can either directly download the results in the .zip archive or navigate themselves into the 'Results page' for further analysis. The 'Results page' is intuitively organized into several panels as described below.

Protein visualization. The wild-type and the mutant structure is interactively visualized in the web browser (Figure 2D) utilizing the Jsmol applet (http://wiki.jmol.org/index.php/JSmol). Users can switch between different protein visualization styles and also highlight selected amino acids in the protein structure. Residues that were included into energy-based mutant are colored in orange, evolution-based mutations are in blue and all other residues are in gray. User selected residues that were not part of any mutant are underlined in red.

Mutant overview. The 'Mutant overview' panel is organized into four tabs (Figure 2A). The first three tabs provide information about mutations included into combined, energy-based and evolution-based mutant. The checkbox, allowing users to visualize the chosen residues in Jsmol applet, can be found in each row together with all data relevant for a given computational approach. The last tab contains the list of all residues in the wild-type structure. While 'wild-type' tab is active, the wild-type structure is visualized in Jsmol applet instead of the mutated one and the user is allowed to introduce user-defined mutations into multiple-point mutant via the 'plus' icon in the last column.

General information. The 'FireProt protocol design' panel provides users with general information about the target protein and the designs constructed by FireProt strategy, such as a number of mutations and estimated change in free energy (Figure 2C).

Mutant designer. The 'Mutant designer' panel allows the user to design own multiple-point mutant by managing mutations divided into energy- and evolution-based subset. If all mutations in the subset have their predicted energy values assigned, a total change in Gibbs free energy is immediately estimated assuming simple additivity. Users can also generate an amino acid sequence from the designed multiple-point mutant that combines mutations included into energy- and evolution-based subsets. All prepared designs can be downloaded in one .zip archive (Figure 2E).

EXPERIMENTAL VALIDATION

The original FireProt strategy was experimentally verified with three proteins (haloalkane dehalogenase DhaA, PDB ID 4E46; γ-hexachlorocyclohexane dehydrochlorinase LinA, PDB ID 3A76; and fibroblast growth factor 2, PDB ID 4OEE) and provided respective stabilization of proteins $\Delta T_m = 25$, 21 and 15°C (Table 1). The original protocol was modified to enable fully automated calculation at the reasonable time, while maintaining high prediction accuracy (Supplementary Table S6). Prediction of eight multiple-point mutants using this modified protocol was validated using the data of FRESCO (44) and identified mutations were compared with another online protein stabilization tool PROSS (17). FireProt and PROSS showed similar predictive power, correctly identifying 29 and 20 potentially stabilizing positions, respectively (Supplementary Table S7).

CONCLUSIONS AND OUTLOOK

FireProt is a web server that provides users with a one-stop-shop solution for the design of thermostable multiple-point mutant proteins. In comparison with the standalone FireProt strategy (16), all default parameters and computational protocols were optimized to increase the calculation speed, while maintaining the prediction accuracy. The designs produced by the FireProt workflow were experimentally verified and thus users can obtain highly reliable thermostable proteins with minimal experimental effort. The server is complemented by an easy-to-use graphical interface that allows users to interactively analyze individual mutations selected as a part of energy- or evolution-based approach together with the ability to design their own multiple-point mutants on top of our robust strategy.

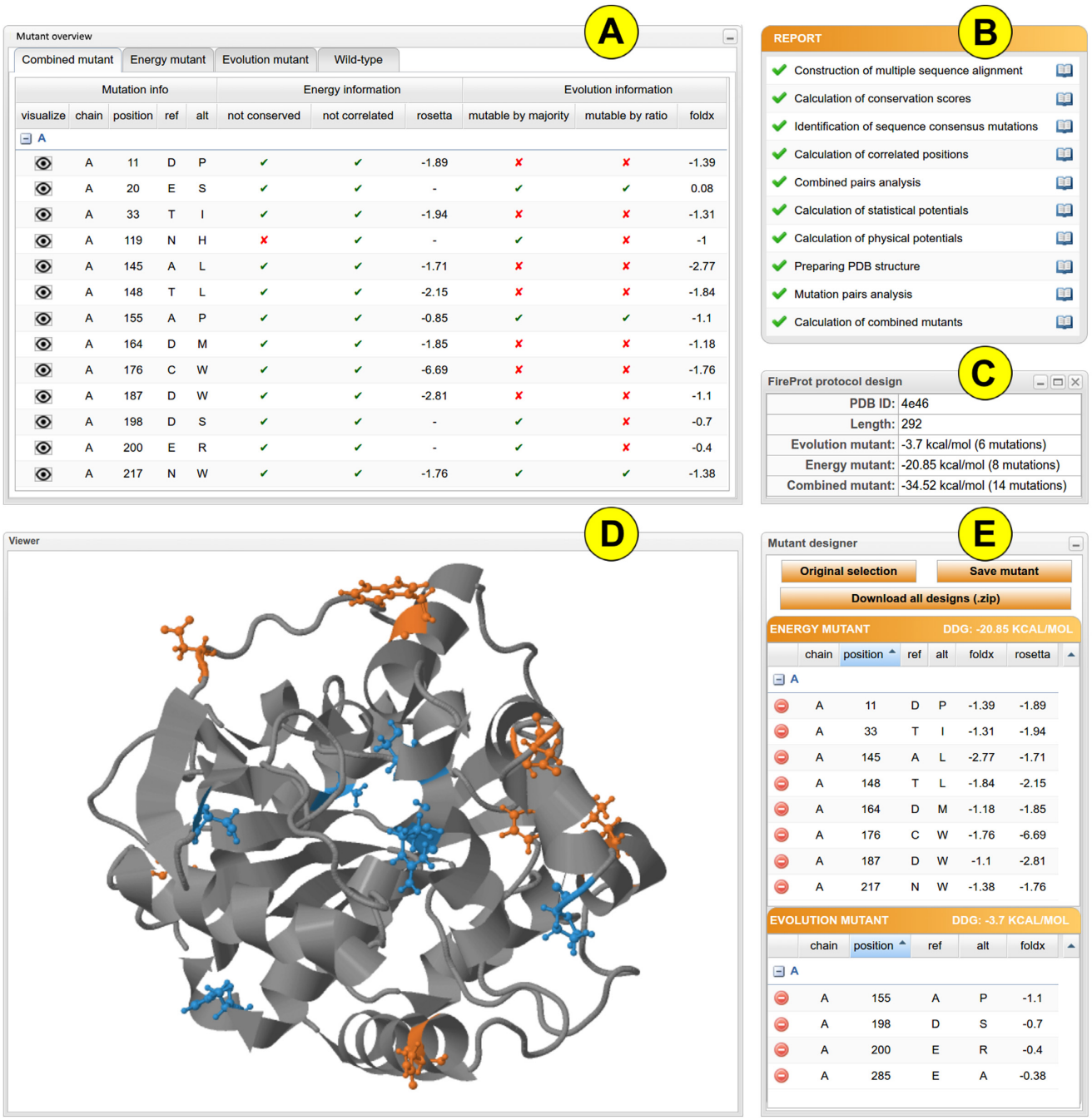

Figure 2. FireProt's graphical user interface showing the results obtained for the haloalkane dehalogenase DhaA (PDB ID: 4e46). (**A**) The 'Mutant overview' panel provides a list of mutations introduced into protein structure. (**B**) The 'Report' panel shows the status of calculation in the individual steps of the computational pipeline. (**C**) The 'Protocol design' panel provides general information about FireProt designs. (**D**) The JSmol ´Viewer´ allows interactive visualization of the protein. (**E**) The 'Mutant designer' panel enables manual adjustment of a new combined mutant.

Table 1. Experimental validation of FireProt strategy

Protein PDB ID	Energy-based mutations	Evolution-based mutations	ΔT_m [°C]
4E46	8	3	+25
3A76	4	3	+21
4OEE	4	2	+15

The automation of the whole procedure makes the process of the design of thermostable proteins accessible to users without any prior expertise in bioinformatics since it eliminates the need to select, install and evaluate tools, optimize their parameters, and interpret intermediate results. However, the energy-based approach of the FireProt strategy depends on the quality of provided protein structure and therefore the prediction accuracy might be compromised in the case of low-resolution structures or homology models.

In the future, we plan to implement new strategies such as a design based on the analysis of correlated positions that would contribute to the construction of the final combined mutant, elimination of highly flexible regions and introduction of disulfide bridges. Also, we plan to equip FireProt with several new filters, e.g. exclusion of the amino acids located in the close neighborhoods of the active sites or the ones participating in oligomerization.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

We would like to thank the developers of PROSS Adi Goldenzweig and Dr Sarel Fleishman (Weizmann Institute of Science, Israel) for providing the data set for validation purposes. Computational resources were supplied by the Ministry of Education, Youth and Sports of the Czech Republic under the Projects CESNET (Project No. LM2015042) and CERIT-Scientific Cloud (Project No. LM2015085) provided within the program Projects of Large Research, Development and Innovations Infrastructures.

FUNDING

This research and open access charge was funded by Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability II [LQ1605, LO1214]; European Regional Development Fund [ELIXIR-CZ LM2015047]; Grant Academy of the Czech Republic [16-06096S]; Brno University Technology [FIT-S-17-3964].

Conflict of interest statement. None declared.

REFERENCES

1. Modarres,H.P., Mofrad,M.R. and Sanati-Nezhad,A. (2016) Protein thermostability engineering. *RSC Adv.*, **6**, 115252–115270.
2. Ferdjani,S., Ionita,M., Roy,B., Dion,M., Djeghaba,Z., Rabiller,C. and Tellier,C. (2011) Correlation between thermostability and stability of glycosidases in ionic liquid. *Biotechnol. Lett.*, **33**, 1215–1219.
3. Wijma,H.J., Floor,R.J. and Janssen,D.B. (2013) Structure- and sequence-analysis inspired engineering of proteins for enhanced thermostability. *Curr. Opin. Struct. Biol.*, **23**, 588–594.
4. Gao,D., Narasimhan,D.L., Macdonald,J., Brim,R., Ko,M.C., Landry,D.W., Woods,J.H., Sunahara,R.K. and Zhan,C.G. (2009) Thermostable variants of cocaine esterase for long-time protection against cocaine toxicity. *Mol. Pharmacol.*, **75**, 318–323.
5. Polizzi,K.M., Bommarius,A.S., Broering,J.M. and Chaparro-Riggers,J.F. (2007) Stability of biocatalysts. *Curr. Opin. Chem. Biol.*, **11**, 220–225.
6. Gray,K.A., Richardson,T.H., Kretz,K., Short,J.M., Bartnek,F., Knowles,R., Kan,L., Swanson,P.E. and Robertson,D.E. (2001) Rapid evolution of reversible denaturation and elevated melting temperature in a microbial haloalkane dehalogenase. *Adv. Synth. Catal.*, **343**, 607–617.
7. Folkman,L., Stantic,B., Sattar,A. and Zhou,Y. (2016) EASE-MM: sequence-based prediction of mutation-induced stability changes with feature-based multiple models. *J. Mol. Biol.*, **428**, 1394–1405.
8. Capriotti,E., Fariselli,P. and Casadio,R. (2005) I-Mutant2.0: predicting stability changes upon mutation from the protein sequence or structure. *Nucleic Acids Res.*, W306–W310.
9. Pires,D.E., Ascher,D.B. and Blundell,T.L. (2014) mCSM: predicting the effects of mutations in proteins using graph-based signatures. *Bioinformatics*, **30**, 335–342.
10. Kellogg,E.H., Leaver-Fay,A. and Baker,D. (2011) Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins*, **79**, 830–838.
11. Yin,S., Ding,F. and Dokholyan,N.V. (2007) Modeling backbone flexibility improves protein stability estimation. *Structure*, **15**, 1567–1576.
12. Dehouck,Y., Grosfils,A., Folch,B., Gilis,D., Bogaerts,P. and Rooman,M. (2009) Fast and accurate predictions of protein stability changes upon mutations using statistical potentials and neural networks: PoPMuSiC-2.0. *Bioinformatics*, **25**, 2537–2543.
13. Guerois,R., Nielsen,J.E. and Serrano,L. (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J. Mol. Biol.*, **320**, 369–387.
14. Gumulya,Y. and Reetz,M.T. (2011) Enhancing the thermal robustness of an enzyme by directed evolution: least favorable starting points and inferior mutants can map superior evolutionary pathways. *ChemBioChem*, **12**, 2502–2510.
15. Bommarius,A.S. and Paye,M.F. (2013) Stabilizing biocatalysts. *Chem. Soc. Rev.*, **42**, 6534–6565.
16. Bednar,D., Beerens,K., Sebestova,E., Bendl,J., Khare,S., Chaloupkova,R., Prokop,Z., Brezovsky,J., Baker,D. and Damborsky,J. (2015) FireProt: energy- and evolution-based computational design of thermostable multiple-point mutants. *PLoS Comput. Biol.*, **11**, e1004556.
17. Goldenzweig,A., Goldsmith,M., Hill,S.E., Gertman,O., Laurino,P., Ashani,Y., Dym,O., Unger,T., Albeck,S., Prilusky,J. *et al.* (2016) Automated structure- and sequence-based design of proteins for high bacterial expression and stability. *Mol. Cell*, **63**, 337–346.
18. Camacho,C., Coulouris,G., Avagyan,V., Ma,N., Papadopoulos,J., Bealer,K. and Madden,T.L. (2009) BLAST+: architecture and applications. *BMC Bioinformatics*, **10**, 421.
19. Suzek,B.E., Wang,Y., Huang,H., McGarvey,P.B. and Wu,C.H. (2015) UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, **31**, 926–932.
20. Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, **26**, 2460–2461.
21. Sievers,F., Wilm,A., Dineen,D., Gibson,T.J., Karplus,K., Li,W., Lopez,R., McWilliam,H., Remmert,M., Soding,J. *et al.* (2011) Fast, scalable generation of high-quality protein multiple sequence alignments using clustal omega. *Mol. Syst. Biol.*, **7**, 539.
22. Capra,J.A. and Singh,M. (2007) Predicting functionally important residues from sequence conservation. *Bioinformatics*, **23**, 1875–1882.
23. Kass,I. and Horovitz,A. (2002) Mapping pathways of allosteric communication in GroEL by analysis of correlated mutations. *Proteins*, **48**, 611–617.
24. Korber,B.T.M., Farber,R.M., Wolpert,D.H. and Lapedes,A.S. (1993) Covariation of mutations in the V3 loop of human immunodeficiency virus type 1 envelope protein: an information theoretic analysis. *Proc. Natl. Acad. Sci. U.S.A.*, **90**, 7176–7180.
25. Lee,B.C. and Kim,D. (2009) A new method for revealing correlated mutations under the structural and functional constraints in proteins. *Bioinformatics*, **25**, 2506–2513.
26. Weigt,M., White,R.A., Szurmant,H., Hoch,J.A. and Hwa,T. (2008) Identification of direct residue contacts in protein–protein interaction by message passing. *Proc. Natl. Acad. Sci. U.S.A.*, **106**, 67–72.
27. Lockless,S.W. and Ranganathan,R. (1999) Evolutionarily conserved pathways of energetic connectivity in protein families. *Science*, **286**, 295–299.
28. Dekker,J.P., Fodor,A., Aldrich,R.W. and Yellen,G. (2004) A perturbation-based method for calculating explicit likelihood of

evolutionary co-variance in multiple sequence alignments. *Bioinformatics*, **20**, 1565–1572.
29. Valencia,A. (2003) Multiple sequence alignments as tools for protein structure and function prediction. *Compar. Funct. Genomics*, **4**, 424–427.
30. Benner,S.A. and Gerloff,D. (1991) Patterns of divergence in homologous proteins as indicators of secondary and tertiary structure: a prediction of the structure of the catalytic domain of protein kinases. *Adv. Enzyme Regul.*, **31**, 121–181.
31. Brenner,S. (1988) The molecular evolution of genes and proteins: a tale of two serines. *Nature*, **334**, 528–530.
32. Cooperman,B.S., Baykov,A.A. and Lahti,R. (1992) Evolutionary conservation of the active site of soluble inorganic pyrophosphatase. *Trends Biochem. Sci.*, **17**, 262–266.
33. Howell,N. (1989) Evolutionary conservation of protein regions in the protonmotive cytochrome b and their possible roles in redox catalysis. *J. Mol. Evol.*, **29**, 157–169.
34. Gobel,U., Sander,C., Schneider,R. and Valencia,A. (1994) Correlated mutations and residue contacts in proteins. *Proteins*, **18**, 309–317.
35. Neher,E. (1994) How frequent are correlated changes in families of protein sequences? *Proc. Natl. Acad. Sci. U.S.A.*, **91**, 98–102.
36. Taylor,W.R. and Hatrick,K. (1994) Compensating changes in protein multiple sequence alignments. *Protein Eng.*, **7**, 341–348.
37. Bava,K.A., Gromiha,M.M., Uedaira,H., Kitajima,K. and Sarai,A. (2004) ProTherm, version 4.0: thermodynamic database for proteins and mutants. *Nucleic Acids Res.*, **32**, D120–D121.
38. Pucci,F., Bourgeas,R. and Rooman,M. (2016) Predicting protein thermal stability changes upon point mutations using statistical potentials: introducing HoTMuSiC. *Scientific Rep.*, **6**, 23257.
39. Amin,N., Liu,A.D., Ramer,S., Aehle,W., Meijer,D., Metin,M., Wong,S., Gualfetti,P. and Schellenberger,V. (2004) Construction of stabilized proteins by combinatorial consensus mutagenesis. *Protein Eng. Des. Select.*, **17**, 787–793.
40. Lehmann,M., Loch,C., Middendorf,A., Studer,D., Lassen,S.F., Pasamontes,L., van Loon,A.P. and Wyss,M. (2002) The consensus concept for thermostability engineering of proteins: further proof of concept. *Protein Eng.*, **15**, 403–411.
41. Pey,A.L., Rodriguez-Larrea,D., Bomke,S., Dammers,S., Godoy-Ruiz,R., Garcia-Mira,M.M. and Sanchez-Ruiz,J.M. (2008) Engineering proteins with tunable thermodynamic and kinetic stabilities. *Proteins*, **71**, 165–174.
42. Sullivan,B.J., Nguyen,T., Durani,V., Mathur,D., Rojas,S., Thomas,M., Syu,T. and Magliery,T.J. (2012) Stabilizing proteins from sequence statistics: the interplay of conservation and correlation in triosephosphate isomerase stability. *J. Mol. Biol.*, **420**, 384–399.
43. Bendl,J., Stourac,J., Sebestova,E., Vavra,O., Musil,M., Brezovsky,J. and Damborsky,J. (2016) HotSpot wizard 2.0: automated design of site-specific mutations and smart libraries in protein engineering. *Nucleic Acids Res.*, **44**, W479–W487.
44. Floor,R.J.1, Wijma,H.J., Colpa,D.I., Ramos-Silva,A., Jekel,P.A., Szymański,W., Feringa,B.L., Marrink,S.J. and Janssen,D.B. (2014) Computational library design for increasing haloalkane dehalogenase stability. *ChemBioChem*, **15**, 1660–1672.

FireProtASR: A Web Server for Fully Automated Ancestral Sequence Reconstruction.

Briefings in Bioinformatics, 22(4), 2021, 1–11

https://doi.org/10.1093/bib/bbaa337
Problem Solving Protocol

FireProt^{ASR}: A Web Server for Fully Automated Ancestral Sequence Reconstruction

Milos Musil, Rayyan Tariq Khan, Andy Beier, Jan Stourac, Hannes Konegger, Jiri Damborsky and David Bednar

Corresponding author: David Bednar, Department of Experimental Biology and RECETOX, Loschmidt Laboratories, Faculty of Science, Masaryk University, 611 37 Brno, Czech Republic. Tel.: +420 605 143 394. E-mail: davidbednar1208@gmail.com

Abstract

There is a great interest in increasing proteins' stability to widen their usability in numerous biomedical and biotechnological applications. However, native proteins cannot usually withstand the harsh industrial environment, since they are evolved to function under mild conditions. Ancestral sequence reconstruction is a well-established method for deducing the evolutionary history of genes. Besides its applicability to discover the most probable evolutionary ancestors of the modern proteins, ancestral sequence reconstruction has proven to be a useful approach for the design of highly stable proteins. Recently, several computational tools were developed, which make the ancestral reconstruction algorithms accessible to the community, while leaving the most crucial steps of the preparation of the input data on users' side. FireProt^{ASR} aims to overcome this obstacle by constructing a fully automated workflow, allowing even the unexperienced users to obtain ancestral sequences based on a sequence query as the only input. FireProt^{ASR} is complemented with an interactive, easy-to-use web interface and is freely available at https://loschmidt.chemi.muni.cz/fireprotasr/.

Key words: ancestral sequence reconstruction; ancestral enzymes; evolution; phylogeny-based analysis; protein stability

Introduction

Proteins are widely used in numerous biomedical and biotechnological applications. Native proteins have mainly evolved under mild intracellular conditions [1]. Therefore, their applicability is often limited in the harsh industrial environments characterized by inhospitable temperature, extreme pH, high pressure or the presence of organic co-solvents. As a result, there is a continuous interest in increasing protein stability. New approaches in the field of protein engineering, such as fluorescence-activated cell sorting and microfluidics, have widened the throughput of

Milos Musil is a bioinformatician at Loschmidt Laboratories, Masaryk University. He carries out his doctoral thesis at the Brno University of Technology, designing and implementing bioinformatics tools for the automatized design of stable proteins.
Rayyan Tariq Khan is a doctoral candidate at Loschmidt Laboratories, Masaryk University. His work is focused on ancestral sequence reconstruction, experimental evolution and design of bioinformatics tools.
Andy Beier is doing his postdoc in protein engineering at the Loschmidt Laboratories, Masaryk University. His main responsibilities are the mutagenesis, production and detailed biochemical and biophysical characterization of enzymes and the development of an ultra-high-throughput assay for dehalogenases.
Jan Stourac is a bioinformatician at Loschmidt Laboratories, Masaryk University. He carries out his doctoral thesis at the Faculty of Informatics, Masaryk University, focusing on the design and implementation of the bioinformatics tools for the analysis of protein tunnels.
Hannes Konegger is a former MSCA fellow, examined the biochemical basis of evolutionary- and structure-based protein engineering methods. He recently turned his field of interest towards microbial ecology and applied bioenergetics.
Jiri Damborsky is a professor of Biochemistry at Masaryk University and group leader at International Clinical Research Center at St. Ann's Teaching Hospital. He is interested in development of software tools for computational enzyme design.
David Bednar is a team leader at Loschmidt Laboratories, Masaryk University. His team is focused on molecular modelling, bioinformatics and development of software for protein engineering.
Submitted: 14 August 2020; **Received (in revised form):** 12 October 2020

directed evolution experiments. However, saturation mutagenesis of all positions and systematic re-combinations of many single-point mutations of the protein of interest is often out of reach.

In the past decades, various computational methods were designed to unburden costly and laborious experimental work by narrowing down the search space for potential stabilizing mutations. Most of those methods can be assigned to one of the three categories: (i) machine learning, (ii) force-field-based predictions and (iii) molecular evolution. Each category has its advantages and shortcomings [2]. Machine-learning methods are able to unearth hidden features and dependencies overreaching the current state of expert knowledge, while still struggling with the insufficient size, quality and diversity of the experimental data, essential for training and validation of statistically significant models. Force-field-based approaches are a robust solution for the prediction of protein stability; however, they rely on the high-resolution protein structures that are available for only a small fraction of the known proteins. Evolution-based approaches do not suffer from these limitations due to the rapid growth of the sequence databases. However, this continuous growth widens the search space and increases noise in the data, requiring laborious and time-demanding manual corrections from the side of the user with expert knowledge of the system of interest. Inexperienced user may not therefore utilize evolution-based methods effectively to obtain accurate and reliable results.

The two most widely used evolution-based methods for stability engineering are ancestral sequence reconstruction (ASR) and consensus design. Both methods start with the multiple-sequence alignment (MSA) of the set of relevant homolog sequences. Consensus design relies on the simple analysis of the conservation of the amino acids on the individual positions in the sequence alignment. As a result, it cannot account for the coevolution of the residues located in the sites responsible for the protein's activity [3] and is utilized mostly as a part of the hybrid workflows [4, 5]. In comparison, ASR goes much further by also considering evolutionary information depicted by the phylogenetic tree. This inclusion of the evolutionary distances inscribed into the phylogenetic tree is mostly negligent at the positions with low Shannon entropy; however, the discrepancies grow stronger with noisy MSA [6]. ASR is a probabilistic method that explores the deep evolutionary history of homolog sequences to reassemble protein's evolutionary trajectory [7]. ASR is able to unearth sequences of the long-extinct genes and organisms from which the current ones evolved and is, therefore, an invaluable tool in the field of evolutionary biology [8, 9]. ASR has also been shown to be a very effective strategy not only for thermostability engineering [10, 11], but also for improving other protein's characteristics such as specificity [12], activity, or expression [13]. Furthermore, ASR was previously proven to be an effective strategy for the stabilization of prokaryotic proteins [10, 11], as well as for the improvement of significantly more complex eukaryotic proteins such as cytochrome P450 [14, 15]. Two main algorithms, maximum-likelihood [16, 17] (ML) and Bayesian inference [18] (BI) were designed to infer ancestral sequence from MSA and phylogenetic tree. Many tools were built over the years to make those algorithms accessible to the community. However, the requirement of the MSA of carefully selected homologs and the rooted phylogenetic tree are still huge limiting steps for the general use of ASR method by the non-expert users.

FireProt^{ASR} addresses those limitations by introducing one-stop-shop solution for the ancestral sequence reconstruction. It covers all steps of ancestral inference including search for homolog sequences, selection of the biologically relevant subset of the sequences, construction of the multiple-sequence alignment, construction and rooting of the phylogenetic tree and finally the ancestral inference with the use of ML. Our computational workflow is fully automated and removes the need for extensive expert knowledge of the system of interest as well as employed bioinformatics tools. Furthermore, a novel algorithm based on the localized weighted back-to-consensus analysis was utilized to resolve an issue of the ancestral gaps reconstruction. Assembled workflow and developed web server were thoroughly validated using: (i) *in-house* laboratory experiments, (ii) detailed comparison with three previously published studies and (iii) a large number of proteins representing structurally and functionally different families. FireProt^{ASR} does not require installation and settings of any software packages as the method is implemented in the interactive web interface freely available at: https://loschmidt.chemi.muni.cz/fireprotasr/.

Methods

Workflow description

The basic workflow of the FireProt^{ASR} method is outlined in Figure 1. To infer ancestral sequences representing all ancestral nodes of the evolutionary tree in a fully automated way, a set of biologically relevant homologous sequences must be collected from genomic databases and reduced to a suitable size (Phase 1). With the initial set of homologous sequences in hand, several state-of-the-art methods are utilized to construct a multiple-sequence alignment and a phylogenetic tree, which are then used to support the inference of ancestral nodes and reconstruction of ancestral gaps (Phase 2). The FireProt^{ASR} workflow requires no user intervention beyond providing a query sequence and (in the case of enzymes) selecting catalytic residues used to identify a biologically relevant set of homologous sequences. However, it is also possible to start a calculation with a user-defined initial set of homologous sequences, MSA, or even a phylogenetic tree instead of a single sequence, thus skipping the first phase of the calculation.

Phase 1: collection of the initial set of homologous sequences

The query sequence of the target protein in plain text or FASTA format is the only input required from the side of the user. Once the query sequence has been uploaded to the server and checked for validity, searches for the catalytic residues are performed automatically using SwissProt [19] and the Catalytic Site Atlas [20]. The user can also specify the catalytic residues by themselves if no/incorrect catalytic residues are found. Once the catalytic residues and query sequence have been specified, an in-house tool called EnzymeMiner [21] is used to collect an initial set of homologous sequences. EnzymeMiner first performs two rounds of PSI-BLAST [22] against the NCBI nr database [23] and then filters out all sequences lacking the designated catalytic residues, thereby ensuring the biological relevance of the remaining homologs. EnzymeMiner searches can yield up to tens of thousands of homologous sequences for large families. If no catalytic residues were selected or provided by the user, BLAST [24] will be used instead of EnzymeMiner, to obtain an initial set of homologous sequences with potentially lower quality.

Next, the FireProt^{ASR} reduces the set of homologous sequences to the required number, which is set to 150 sequences by default. Several filters are applied during this process. First, all homologs

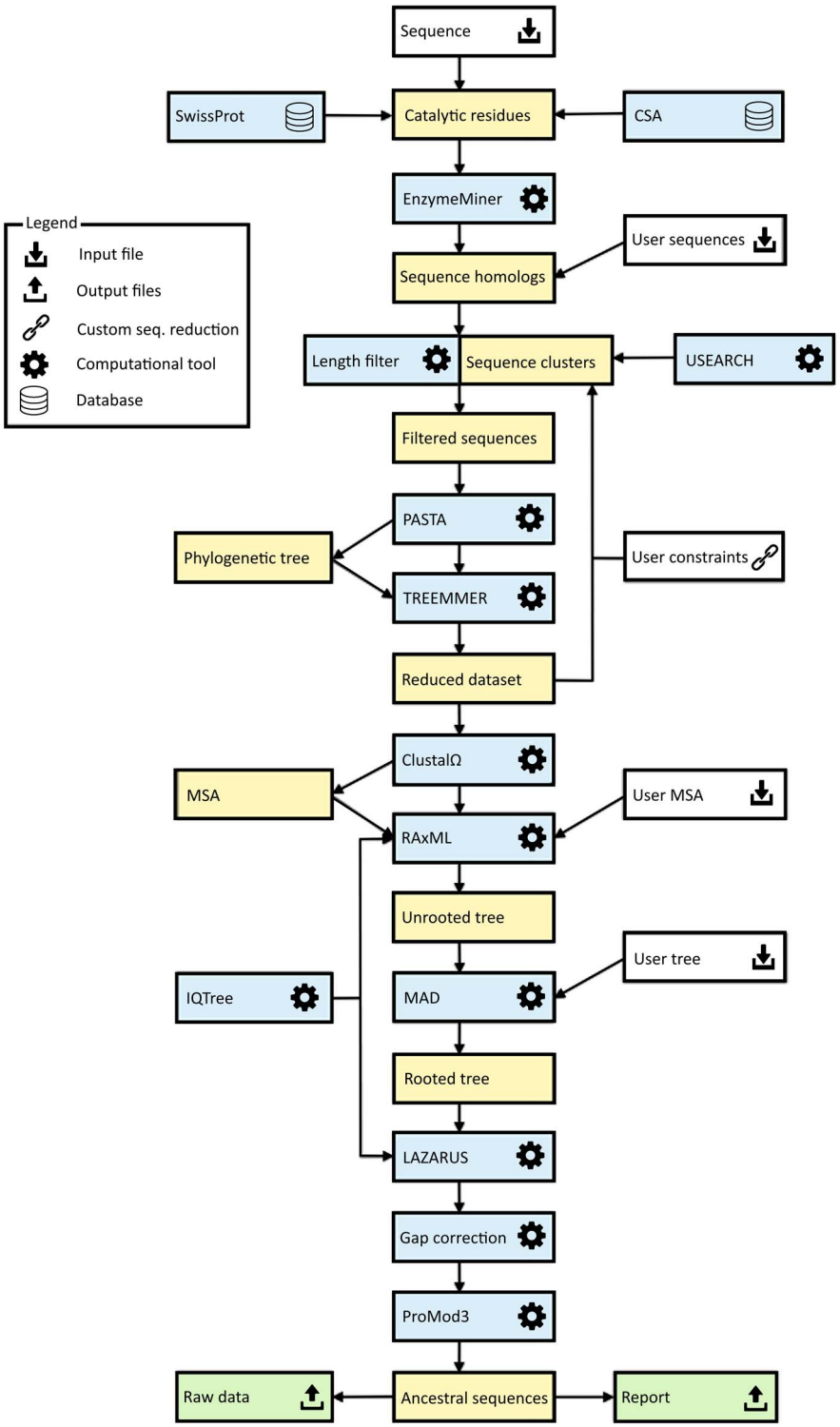

Figure 1. Workflow diagram for the FireProt^{ASR} method. The workflow has two phases: (1) collection of the initial set of homologous sequences and (2) ancestral sequence reconstruction. Colour coding: yellow denotes intermediate results and blue denotes computational tools. Grey and green denote inputs and outputs of the calculations, respectively.

with sequence lengths 20% higher or lower than that of the query sequence are excluded from the initial set. This sequence length normalization is done to remove potential outliers that could lead to a construction of a noisy MSA with many gaps. Second, all homologs whose sequence identity to the query falls outside a certain range are removed from the initial set. By default, the upper and lower similarity limits are set to 90 and 30%, respectively. This step ensures that the phylogenetic tree is unbiased towards the query sequence while removing distant homologs that would degrade the quality of the sequence

alignment. Third, USEARCH [25] is used to cluster the remaining sequences with 90% sequence identity, and a single sequence is randomly selected from each cluster.

Applying these filters produces a diverse set containing hundreds to thousands of homologous sequences. An initial phylogenetic tree is quickly constructed with the PASTA software suite [26], using MAFFT [27] and the swift neighbour-joining algorithm implemented in FastTree 2.0 [28]. The resulting phylogenetic tree is then forwarded to Treemmer [29], which iteratively prunes leaves from the input tree until a specific number of leaves remains, while minimizing the loss of genetic diversity. The pruned tree is then displayed to the user via the interactive user interface, allowing the user to choose to exclude selected branches or even whole subtrees of the phylogenetic tree from further calculations.

Phase 2: ancestral sequence reconstruction

In the second phase, the ancestral sequences are inferred from the initial set of up to 150 homologs approved by the user. To begin with, a new MSA is constructed from the reduced set of homologous sequences. For this task, ClustalΩ [30] is utilized by default, but other methods will be available in upcoming versions of FireProt^{ASR}. For inference of the final phylogenetic tree, the best-fitting evolutionary matrix must be selected. This is done using one of the modules of the IQTREE package [31]. Alternatively, if the user prefers a specific evolutionary matrix for the biological system of interest, the appropriate model and all the relevant modifiers can be specified manually when setting up the calculation.

The evolutionary model and its parameter settings along with the MSA are then forwarded into RAxML [17], which is used to construct a robust phylogenetic tree. By default, fifty bootstraps are performed at the start of the maximum-likelihood search; since no outgroup is provided, the resulting phylogenetic tree is unrooted. Automated outgroup sequence selection is not straightforward, especially for prokaryotic proteins due to the high frequency of horizontal gene transfers. Rooting of the tree is therefore performed using a minimal ancestor deviation algorithm, which was shown to achieve comparable levels of accuracy to outgroup rooting in trees describing the evolution of eukaryotes, and to surpass both outgroup and midpoint rooting in the case of prokaryotes [32].

The MSA constructed with ClustalΩ, the selected evolutionary model, and the rooted phylogenetic tree from RAxML are used as inputs for the Lazarus method [33], which is implemented using the PAML software package [16]. The Lazarus method was re-implemented for FireProt^{ASR} to enable calculations to be performed without specifying outgroup. Consequently, ancestral sequences of all ancestral nodes are parsed from their posterior probabilities and provided to users in separate files in FASTA format. Additionally, BLASTp [24] is used to search for a template in the PDB database [34], and a model structure of the query sequence is constructed by homology modelling using the ProMod3 program [35]. This model is shown in the web interface to allow users to visualize the differences between the query sequence and the selected ancestor.

Finally, due to the large number of undesirable ancestral gaps inserted into ancestral sequences by Lazarus, a novel algorithm for ancestral gap reconstruction was designed for use in FireProt^{ASR}. This algorithm is based on the principle of localized weighted back-to-consensus because consensus analysis has proven to be an effective approach for increasing proteins' thermal stability [36–38]. To begin with, each terminal node of the phylogenetic tree is assigned a binary vector of length equal to the length of the corresponding sequence in the MSA. Each position in this vector is assigned a value of −1 or 1, indicating the presence of a gap or standard amino acid, respectively, at the corresponding position of the relevant sequence. On moving from the terminals towards the root of the tree, the probability of a gap in ancestral node A_n at position i is calculated as $A_{n_i} = \frac{A_{k_i} * t_1 * + A_{l_i} * t_2}{t_1 + t_2}$, where A_k, A_l are the child nodes of A_n and t_1, t_2 are the evolutionary distances between A_n and its child nodes. Taking t_3 to be the evolutionary distance between A_n and its parental node, its value can be updated based on the values of t_1 and t_2 as follows: $t_{3\_new} = t_3 + \frac{t_1 + t_2}{2}$. This new value is computed before proceeding with the calculation for the parental node; its use increases the relative impact of well-branched subtrees and therefore limits the impact of lone sequences and small subtrees compared to that of well-represented ones. Finally, ancestral sequences are reconstructed based on the scores in the corresponding vector. Positions with values lower than 0 are assigned as gaps, and the remaining amino acids are selected based on their posterior probabilities as estimated by Lazarus. The nature of inconclusive positions with scores in the interval $<-0.1, 0.1>$ is determined based on the frequencies of gaps in the global alignment and the state of the parental node. To include the ancestral gap, frequencies of gaps in the global alignment should reach over 60%, or over 40% if the ancestral gap is present in the parental node sequence. The model case for a single position in the sequence alignment is shown in Figure 2.

Experimental validation

The workflow was experimentally validated using haloalkane dehalogenases as a model enzyme. This enzyme was selected as a typical representative of the α/β superfamily, counting over 100 000 proteins. The sequence of the haloalkane dehalogenase DhaA (UniProt ID P0A3G2) was used as the sole input for the calculation. Six different ancestral sequences were selected and experimentally characterized.

Chemicals and growth media

1-bromobutane and LB medium were purchased from Sigma-Aldrich Co. (St. Louis, MO, USA). IPTG was purchased from Duchefa Biochemie B.V. (Haarlem, The Netherlands). All chemicals used in this work were of analytical grade.

Expression in *Escherichia coli* BL21 (DE3)

Escherichia coli Dh5α cells were obtained from Invitrogen and *Escherichia coli* BL21 (DE3) from New England Biolabs. The genes for the ancestral dehalogenases were synthesized and subcloned into the expression vector pET21b. The generated plasmids were transformed into chemo-competent *E.coli* BL21 (DE3) cells. Obtained colonies were used to prepare precultures by inoculation into 10 ml of LB medium (with 100 µg/ml ampicillin) followed by overnight incubation at 37°C and 180 rpm. For expression of each variant, 1 l of LB medium supplemented with 100 µg/ml ampicillin was inoculated with 5 mL of the appropriate pre-culture (1/200). The flasks were incubated at 37°C and 180 rpm until OD_{600} 0.6–0.8 was reached, then incubated at 20°C for 30 min. β-D-1-thiogalactopyranoside (IPTG, 0.2 mM) was then added for induction, and the culture was incubated at 20°C and 180 rpm overnight. Finally, the culture was harvested by centrifugation at 4500 × g, 4°C for 15 min, after which the cell pellets were frozen at −80°C until further use.

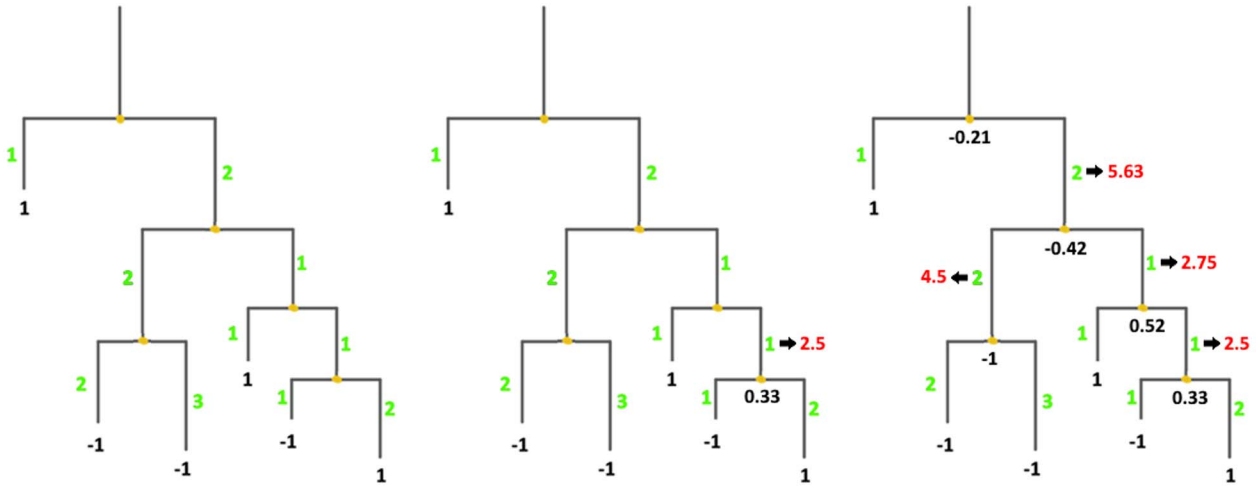

Figure 2. Ancestral gaps reconstruction algorithm. Green colour denotes the initial branch lengths of the phylogenetic tree. Black numbers indicate the values of the vectors of the terminal and the ancestral sequences at the given position in the multiple sequence alignment. Red values show the modified branch lengths that are updated after the calculation of the underlying ancestral node.

Protein purification

The cell pellets were suspended in 50 ml of equilibration buffer (20 mM phosphate buffer pH 7.5 containing 0.5 M NaCl and 10 mM imidazole) and disrupted by sonication with a Hielscher UP200S ultrasonic processor (Hielscher, Germany) four times for 4 min each. Disrupted cells were centrifuged at 13 000 × *g* and 4°C for 1 h (Laborzentrifugen, Germany). The crude extract was then collected, filtered and loaded onto a Ni-NTA Superflow Cartridge (Qiagen, Germany) in equilibration buffer. Unbound and weakly bound proteins were washed out using increasing imidazole concentrations. The target enzyme was eluted with purification buffer containing 300 mM of imidazole. The eluted protein was dialyzed three times overnight against 50 mM of phosphate buffer (pH 7.5), after which its purity was checked by SDS–polyacrylamide gel electrophoresis (SDS–PAGE). About, 15% polyacrylamide gels were stained with Instant Blue (Fluka, Switzerland). Protein concentrations were determined by NanoDrop (Sigma-Aldrich, USA). The enzymes were lyophilized using a vacuum pump system for long-term storage.

Circular dichroism (CD) spectroscopy

CD spectra were recorded at 20°C using a spectropolarimeter Chirascan (Applied Photophysics, United Kingdom). Data were collected from 190 to 260 nm, at 100 nm/min with a 1-s response time and 1-nm bandwidth using a 0.1-cm quartz cuvette. Each spectrum shown is the average of five individual scans and was corrected for absorbance caused by the buffer. Collected CD data were expressed in terms of the mean residue ellipticity (Θ_{MRE}), which was calculated using the equation:

$$\Theta_{MRE} = \frac{\Theta_{obs} \cdot M_w \cdot 100}{n \cdot c \cdot l}$$

where Θ_{obs} is the observed ellipticity in degrees, M_w is the protein molecular weight, n is number of residues, l is the cell path length, c is the protein concentration (0.2 mg/ml) and the factor 100 originates from the conversion of the molecular weight to mg/dmol.

Thermal denaturation

Thermal unfolding was followed by monitoring the ellipticity at 224 nm over the temperature range of 20–94°C, with a resolution of 0.1°C at a heating rate of 1°C/min. Recorded thermal denaturation curves were roughly normalized to represent signal changes between approximately 1 and 0 and fitted to sigmoidal curves using Origin 6.1 (OriginLab Corporation, USA). The melting temperature (T_m) was evaluated as the midpoint of the normalized thermal transition.

Enzymatic haloalkane dehalogenase activity

Dehalogenation activity was assayed using the colorimetric method of Iwasaki *et al.* [49]. The release of halide ions was analyzed spectrophotometrically at 460 nm using an Eon microplate reader (BioTek, USA) after reaction with mercuric thiocyanate and ferric ammonium sulfate. The reactions were performed at 37°C in 25-ml Reacti Flasks closed with Mininert Valves. The reaction mixtures consisted of 10 ml 100 mM glycine buffer (pH 8.6) and 10 µl of the substrate 1-bromobutane. Reactions were initiated by adding the enzyme to a final concentration of 0.01 (DhaA 172Loc), 0.0065 (DhaA 172Glob), 0.0052 (DhaA 230Glob), 0.028 (DhaA 238Loc) or 0.014 mg/ml (DhaA 238Glob). Reactions were monitored by withdrawing 1 ml of samples from the reaction mixture after 0, 5, 10, 15, 20 and 30 min. The samples were immediately mixed with 0.1 ml of 35% nitric acid to stop the reaction. Dehalogenation activities were quantified as rates of product formation over time. Each activity was measured in three independent replicates.

Enzymatic luciferase activity

Luminescence activity measurements were performed with a FLUOstar OPTIMA Microplate reader (BMG Labtech, Germany) using coelenterazine as the substrate at 37°C. A 25 µl of sample of purified enzyme at a concentration of about 1 mg/ml was placed into a microtiter plate well. After baseline collection for 10 s, the luminescence reaction was initiated by adding 225 µl of 8.8 µM coelenterazine in reaction buffer (100 mM potassium phosphate buffer, pH 7.5). Luminescence was recorded for 72.5 s,

and each sample was measured in at least three independent experiments. The areas of the resulting luminescence intensity peaks in relative luminescence units (RLU) were converted into values in units of RLU/mg/s.

Results

Web server input

The only required input to the web server is a query sequence of the target protein in plain text or FASTA format. Alternatively, one can upload a FASTA file containing an initial set of sequence homologs or a multiple sequence alignment (MSA). Rooted and unrooted phylogenetic trees in the standard Newick format can also be provided. When performing calculations in basic mode, only the table containing the essential residues is available to the user. Essential residues are identified automatically by searching in SwissProt [19] and mCSA [20]. However, the initial selection can be changed by the user. The default values and settings of individual computational tools are optimized to provide reliable results for most systems. Operating in advanced mode expands the list of modifiable parameters to include those related to: (i) the thresholds of the homolog identity filters and sequence clustering, (ii) selection of the evolutionary model and (iii) construction of the phylogenetic tree. Advanced mode allows experts to fine-tune the calculation's parameters based on the studied biological system, which may be useful when dealing with particularly small or large protein families.

Selection and reduction

Upon submission, a unique identifier is assigned to each job to track the calculation. The 'calculation browser' informs the user about the status of the individual steps in the ancestral sequence reconstruction workflow. Once the first phase of the job is finished, the initial phylogenetic tree is displayed to the user using a strongly updated adaptation of PhyloTree library (Figure 3A) [39], together with the table of removed sequences (Figure 3B). By clicking on the individual leaves of the phylogenetic tree, the user can exclude selected sequences from future calculations. Furthermore, whole subtrees can be removed by choosing this option in the menu of the selected ancestral node. The MSA of the homologous sequences can be also visualized by switching to the multiple sequence alignment tab. This mode is intended for the expert users with the greater knowledge of the system of interest as it allows for the removal of the noise and outliers from the initial set of homolog sequences. If the expert mode is utilized, it is recommended to exclude the sequences that do not share the function similar to the query protein or that cause a significant disturbance in the MSA.

Web server output

The calculation's progress can be tracked in the 'calculation browser' similarly to the selection step. Once finished, users can either download the results in the zipped archive directly from the calculation page or navigate to the 'Result page' for further analysis. The 'Result page' is organized into several panels allowing users to interactively visualize and design ancestral enzymes.

Protein visualization

The homology model of the query protein predicted by ProMod3 is interactively visualized in the web browser using the JSmol applet [40] (Figure 3D). Users can switch between different visualization styles such as backbone, wireframe or cartoon and change the quality of the visualized structure. It is also possible to visualize the differences between the query and the selected ancestral sequence on the modelled protein structure: substitutions and deletions are shown in blue and red, respectively, while insertions are indicated by regions between red and yellow residues.

Ancestral tree panel

The 'ancestral panel' shows the final phylogenetic tree constructed by RAxML [17] along with further information about the precalculated ancestral sequences (Figure 3E). By selecting any of the ancestral nodes, it is possible to either (i) visualize the differences between a wild-type protein and the selected ancestor node on the protein structure or (ii) open a new window providing an overview of the posterior probabilities for individual amino acids in the sequence of the selected ancestor (Figure 3G). Posterior probabilities are shown in the bar-styled sequence logo together with the percentages for each considered amino acid, and each bar is expanded with information about the charge and hydrophobicity of the most probable amino acids. The bar representation was in part derived from the SequenceLogo library [41]. The user can edit the ancestral sequence and store it as a new user-defined ancestor (Figure 3F). This option is useful for the experts with more in-depth knowledge of the system of interest and allows to force some specific mutations, e.g., the mutations with the previously known effect on proteins stability, into the constructed ancestral sequence. It can also be used to bring some biological insight into the positions with noisy posterior probabilities. Furthermore, the ancestral sequences' MSA can be visualized in the multiple sequence alignment tab for further analysis.

Sequence designer

The 'Sequence designer' panel allows users to manage and edit user-defined ancestral sequences. Additionally, new sequences can be created by modifying existing custom ancestors (Figure 3C). Differences between the query sequence and custom ancestors can also be visualized on the protein structure in this panel. All prepared designs can be downloaded in one zipped archive together with the original ancestors and the structure prepared by homology modelling.

Web server experimental validation

In one of our previous studies, we have presented experimental characterizations of six inferred ancestral proteins from haloalkane dehalogenase subfamily II [10]. Relative to their contemporary counterparts, these ancestral proteins exhibited higher thermal stability (by 8–24°C), improved yields and broadened substrate specificity. Those ancestral sequences were reconstructed by clustering an initial set of homologous sequences that was reduced by inspection in the sequence-editing program BioEdit [42]. A multiple sequence alignment was then manually curated using a structure-guided alignment of eight proteins from HLD-II and poorly conserved regions were removed from the alignment. The topology of the phylogenetic tree was optimized by subtree pruning and re-grafting, and the tree's root was established using outgroup selected on the basis of expert judgement. Finally, the ancestral sequences and positioning of gaps were refined by manual inspection.

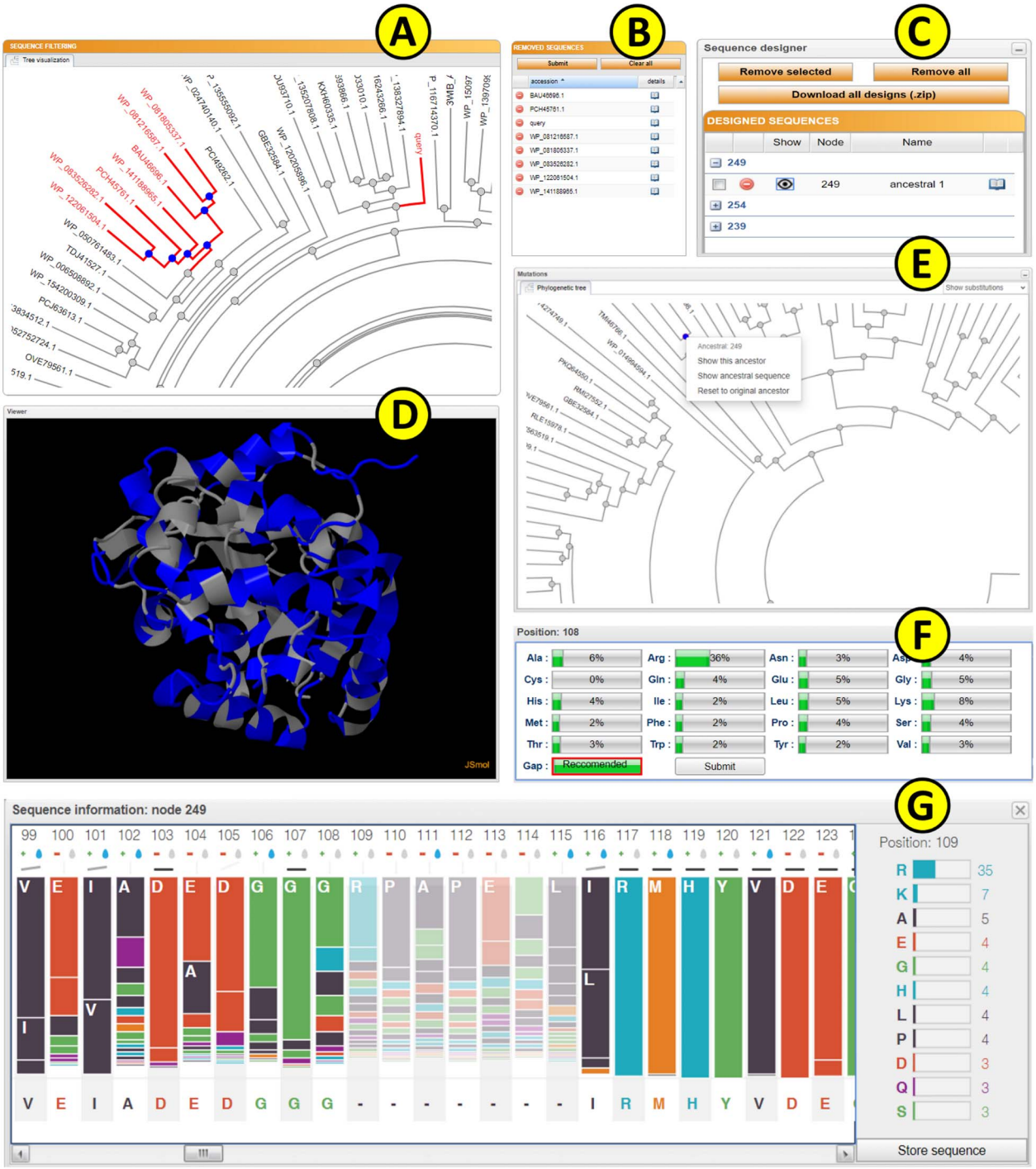

Figure 3. The FireProt^{ASR} graphical user interface showing results obtained for the haloalkane dehalogenase DhaA (UniProt ID P0A3G2, PDB ID 4E46). **(A)** The sequence-filtering panel allows users to exclude selected branches from the calculation. **(B)** The reduction table shows the list of removed sequences. **(C)** The sequence designer allows users to download and edit ancestral sequences. **(D)** The JSmol viewer provides interactive protein visualization. **(E)** The mutations panel contains all designed ancestral sequences in the ancestral tree. **(F)** The edit window enables amino acid substitutions at individual positions. **(G)** The sequence information window shows detailed information on selected ancestral sequences.

As part of the validation of FireProt^{ASR}, we tried to replicate these results by using the sequence of haloalkane dehalogenase DhaA (UniProt ID P0A3G2) as the only input query. All steps of the calculation, including homologous sequence selection, multiple sequence alignment construction, phylogenetic rooting and ancestral reconstruction were carried out automatically. Three pairs of ancestral sequences were selected, each pair containing one 'global' and one 'local' ancestral node (Figure 4A). Global ancestor (Glob) represents ancestral sequence obtained directly from the fully automated workflow, while local ancestor

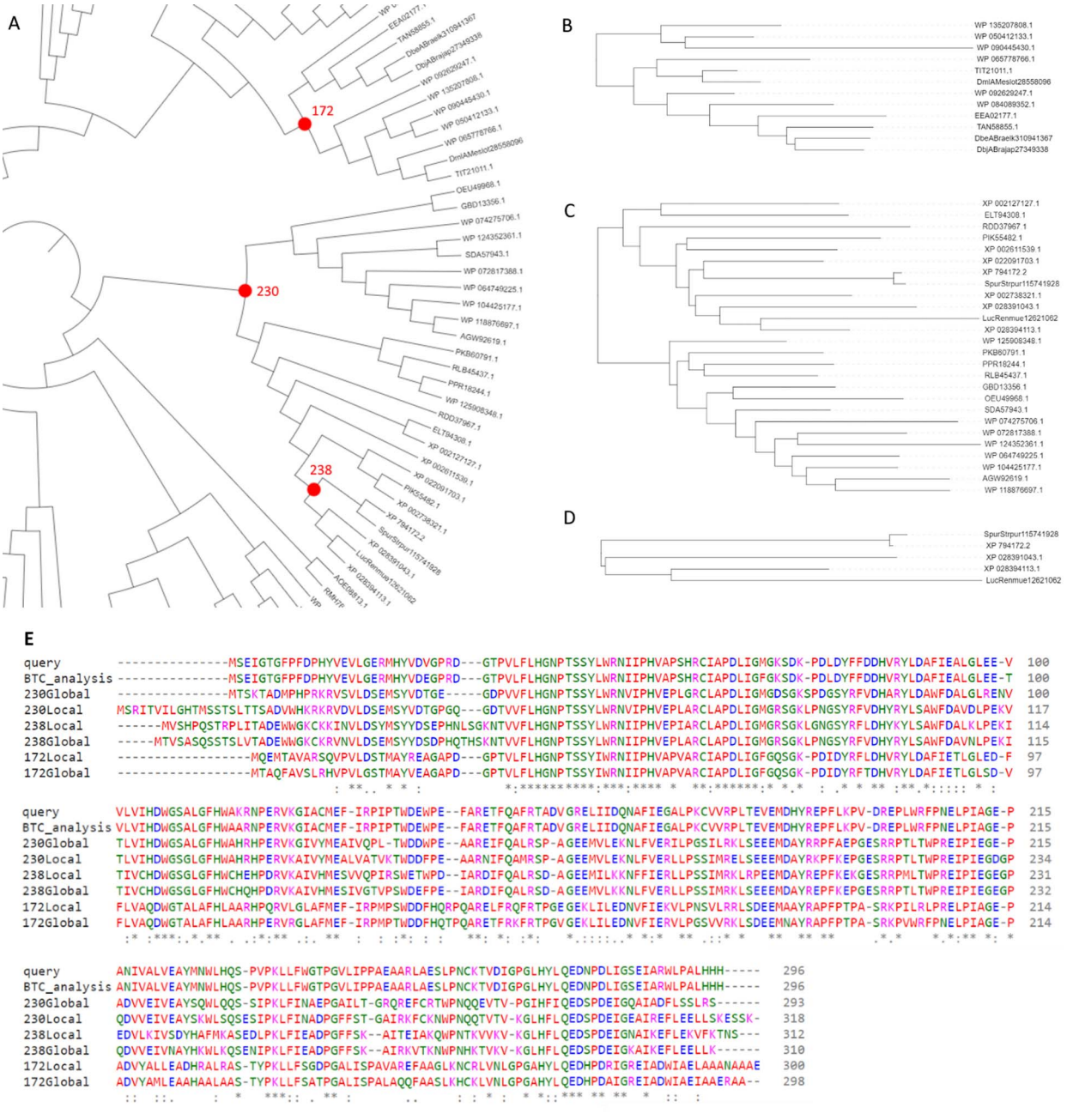

Figure 4. Results provided by the FireProtASR workflow using haloalkane dehalogenase DhaA as an input query. (**A**) Phylogenetic tree of the HLD-II constructed by the FireProtASR strategy with indicated three global ancestors reconstructed within this study. (**B**) Phylogenetic tree for the local ancestor of the ancestral node 172. (**C**) Phylogenetic tree for the local ancestor of ancestral node 230. (**D**) Phylogenetic tree for the local ancestor of ancestral node 238. (**E**) Multiple sequence alignment comparing the query sequence with the suggested ancestral sequences and the result of the back-to-consensus analysis.

(Loc) was constructed by carrying out FireProt^{ASR} workflow for a second time using only the sequences included in the subtree beneath the selected ancestral node. Local ancestor therefore represents a root of a phylogenetic tree constructed from only the sequences most relevant to the selected ancestral node. Node 238 (Figure 4D) is an ancestor of only five leaves and was selected because of its close proximity to luciferase and dehalogenase, providing a fair comparison to the previously published ancestors. Similar comparison can be also achieved with node 172 (Figure 4B), having several stable dehalogenases in its progeny. Finally, node 230 (Figure 4C) was highlighted as a more distant ancestor of both luciferase and dehalogenase. No pruning, curation or re-grafting was performed in the process. Selected ancestral sequences were then subjected to the experimental validation. MSA of the query protein, selected ancestors, and the sequence provided by executing back-to-consensus analysis is attached in Figure 4E.

Although the selected sequences have high implied sequence similarity (92–97%) with the inferred ancestors, experimental validation showed that the ancestors' thermal stability was 20–26°C higher than that of wild-type DhaA (Table 1). The ancestral proteins also exhibited high expressibility, solubility, yields and

Table 1. Characteristics of reconstructed and experimentally characterized ancestral haloalkane dehalogenases

Protein code	Expression (% of total protein)	Solubility (%)	Yield (mg/l)	T_m (°C)	HLD act. (μmol/mg·s)	LUC act. (RLU/mg·s)
DhaA wt	17	83.1	91.1	50.56 ± 2.4	0.032 ± 0.0059	n.a.
DhaA 172Loc	23	85.5	74.9	71.60 ± 0.7	0.038 ± 0.0002	1.41 ± 0.26
DhaA 172Glob	21	65.2	88.2	70.04 ± 1.5	0.061 ± 0.0045	n.a.
DhaA 230Loc	20	n.d.	n.d.	n.d.	n.d.	n.d.
DhaA 230Glob	23	84.8	108.5	72.14 ± 0.4	0.061 ± 0.0118	n.a.
DhaA 238Loc	23	63.2	74.9	70.36 ± 0.6	0.014 ± 0.0021	353.5 ± 14.58
DhaA 238Glob	19	83.3	94.4	76.19 ± 0.2	0.030 ± 0.0012	3.18 ± 0.33

Notes: DhaA, haloalkane dehalogenase from *Rhodococcus rhodochrous* NCIMB 13064; wt, wild type; Loc, ancestral protein inferred from local alignment; Glob, ancestral protein inferred from global alignment; T_m, melting temperature; HLD act., haloalkane dehalogenases activity; LUC act., luciferase activity; n.d., not determined due to poor solubility of this protein; n.a., not active under tested conditions.

catalytic activity. Moreover, inference based on both haloalkane dehalogenases and luciferases led to the discovery of the very interesting enzyme ancHLD-Rluc, which exhibits dual dehalogenase and monooxygenase activity. This experimental validation provides direct experimental evidence of the good functionality and reliability of the fully automated version of FireProt^{ASR}.

Additionally, results obtained using FireProt^{ASR} were thoroughly and quantitatively compared to three previously published experimental studies. For this purpose, Euclidean distance [43], and the Subtree prune and regraft distance [44] were calculated to compare the trees obtained from the FireProt^{ASR} and published literature. The two trees were also graphically compared using the Jaccard index utilizing ColorBrewer [45] scheme. Detailed comparison of all three experimental studies with the results produced by FireProt^{ASR} server is attached in Supplementary Data 1–3, available online at https://academic.oup.com/bib. Finally, the robustness and reliability of the FireProt^{ASR} server was tested using 60 diverse proteins from various protein families (see Supplementary Data 4 available online at https://academic.oup.com/bib).

Discussion

ASR has been shown to be a very effective strategy for the protein thermostability engineering and as such was implemented in various computational tools using maximum-likelihood (FastML [46], RaxML [17], Ancestors [47]) or Bayesian inference (HandAlign [48], MrBayes [18]) methods. However, a significant limitation of those methods is that they require complex input data to be uploaded by the users. Those requirements are reaching from a simple set of homolog sequences to the MSA or even rooted phylogenetic tree, leaving the most crucial and laborious parts of the calculation in the hands of the users. Non-expert users without the deep knowledge of the bioinformatics tools and the system of interest are therefore hindered from the successful use of the ASR method.

FireProt^{ASR} is a web server that aims to provide users with one-stop-shop solution for the ancestral sequence reconstruction. FireProt^{ASR} requires minimal input from the users, and the whole calculation can be processed from a single protein sequence, set of homolog sequences, MSA and phylogenetic tree. All steps of the calculation, including the search for biologically relevant homolog sequences, dataset reduction and the ancestral reconstruction are automated. Moreover, a novel algorithm based on localized weighted back-to-consensus analysis is implemented to resolve an issue with ancestral gap reconstruction. FireProt^{ASR} web server is also complemented by an easy-to-use web interface that allows users to interactively analyze sequences of the individual ancestral nodes together with the ability to design their own ancestral sequences based on the posterior probabilities of the existing nodes.

The robustness and reliability of the results produced by the FireProt^{ASR} workflow was evaluated by experimental characterization of six ancestral sequences of haloalkane dehalogenase from HLD-II subfamily. With the exception of the local variant of the ancestral node 230, all designed ancestral sequences are soluble and also retain high expressibility and yields on the levels comparable to the DhaA wild type. However, the thermal stability has increased by over 20°C and global variants 172 and 230 have also increased the HLD activity by two-fold. Increase in HLD activity cannot be observed in the constructed local variants that utilize smaller subsets of homolog sequences, and thus only a limited amount of evolutionary information. This would encourage the usage of the global variants for the design of highly stable and active proteins. However, more focused view using a localized variants of the ancestral nodes can provide some useful results as can be observed in the local variant of the node 238 that shows both dehalogenase and monooxygenase activity. High thermal stabilization was also achieved in those variants.

Finally, the results provided by the FireProt^{ASR} web server are consistent with the designs presented in the published literature as the fully automatized designs obtained by FireProt^{ASR} method maintain high sequence similarity (>90%) with the manually designed and curated ancestors. Finally, the comprehensive analysis of approximately 60 different proteins from various protein families have proven the robustness and reliability of the presented method.

The full automation of the FireProt^{ASR} method eliminates the need to select, install and evaluate individual tools, optimize their parameters and interpret intermediate results. Together with its general applicability for a wide range of protein families, FireProt^{ASR} makes the procedure of ancestral reconstruction accessible to the users without any prior expertise in bioinformatics, and the intuitive web interface allows for a further analysis utilizing both sequence and structural information.

Key Points

- FireProt^{ASR} is a web service for a fully automated design of stable proteins using ancestral sequence reconstruction and is accompanied by an interactive and easy-to-use interface.

- FireProt^{ASR} allows users to utilize ancestral reconstruction without prior knowledge of the necessary bioinformatics tools and the biological system.
- The robustness and reliability of the FireProt^{ASR} method were thoroughly tested by both laboratory experiments and by comparing predictions with the results published in scientific literature.
- Laboratory characterization of the ancestral designs showed up to 26°C improvement in thermostability and some of the proteins poses even dual catalytic activity.

Data availability

All data validating the robustness and accuracy of our service are available in the Supplementary materials 1-4. Web service and tutorials are freely available at https://loschmidt.chemi.muni.cz/fireprotasr/.

Supplementary Data

Supplementary data are available online at *Briefings in Bioinformatics*.

Funding

Czech Ministry of Education (CZ.02.1.01/0.0/0.0/17_043/0009632, 857560, CZ.02.1.01/0.0/0.0/16_026/0008451); the Czech Grant Agency (20-15915Y); the Technology Agency of Czech Republic (TH02010219); Brno University of Technology (FIT-S-20-6293); and the European Commission (720776, 814418, 722610) and Marie Curie@MUNI (CZ.02.2.69/0.0/0.0/19_074/0012727). Computational resources were supplied by the project 'e-Infrastruktura CZ' (LM2018140) and ELIXIR (LM2015047). This project has received funding from the European Union's Horizon 2020 research and Innovation programme. The article reflects the author's view and the Agency is not responsible for any use that may be made of the information it contains.

References

1. Modarres HP, Mofrad MR, Sanati-Nezhad A. Protein thermostability engineering. *RSC Adv* 2016;**6**:115252–70.
2. Musil M, Konegger H, Hon J, *et al*. Computational design of stable and soluble biocatalysts. *ACS Catal* 2019;**9**:1033–54.
3. Hendrikse NM, Charpentier G, Nordling E, *et al*. Ancestral diterpene cyclases show increased thermostability and substrate acceptance. *FEBS J* 2018;**285**:4660–73.
4. Musil M, Stourac J, Bendl J, *et al*. FireProt: web server for automated design of thermostable proteins. *Nucleic Acids Res* 2017;**45**:W393–9.
5. Goldenzweig A, Goldsmith M, Hill SE, *et al*. Automated structure- and sequence-based design of proteins for high bacterial expression and stability. *Mol Cell* 2016;**63**:337–46.
6. Risso VA, Gavira JA, Gaucher EA, *et al*. Phenotypic comparisons of consensus variants versus laboratory resurrections of Precambrian proteins. *Proteins Struct Funct Bioinforma* 2014;**82**:887–96.
7. Hochberg GKA, Thornton JW. Reconstructing ancient proteins to understand the causes of structure and function. *Annu Rev Biophys* 2017;**46**:247–69.
8. Bickelmann C, Morrow JM, Du J, *et al*. The molecular origin and evolution of dim-light vision in mammals. *Evolution* 2015;**69**:2995–3003.
9. Hobbs JK, Prentice EJ, Groussin M, *et al*. Reconstructed ancestral enzymes impose a fitness cost upon modern bacteria despite exhibiting favourable biochemical properties. *J Mol Evol* 2015;**81**:110–20.
10. Babkova P, Sebestova E, Brezovsky J, *et al*. Ancestral haloalkane dehalogenases show robustness and unique substrate specificity. *Chembiochem* 2017;**18**:1448–56.
11. Risso VA, Gavira JA, Sanchez-Ruiz JM. Thermostable and promiscuous Precambrian proteins. *Environ Microbiol* 2014;**16**:1485–9.
12. Wheeler LC, Lim SA, Marqusee S, *et al*. The thermostability and specificity of ancient proteins. *Curr Opin Struct Biol* 2016;**38**:37–43.
13. Zakas PM, Brown HC, Knight K, *et al*. Enhancing the pharmaceutical properties of protein drugs by ancestral sequence reconstruction. *Nat Biotechnol* 2017;**35**:35–7.
14. Bart AG, Harris KL, Gillam EMJ, *et al*. Structure of an ancestral mammalian family 1B1 cytochrome P450 with increased thermostability. *J Biol Chem* 2020;**295**:5640–53.
15. Gumulya Y, Baek J-M, Wun S-J, *et al*. Engineering highly functional thermostable proteins using ancestral sequence reconstruction. *Nat Catal* 2018;**1**:878–88.
16. Yang Z. PAML 4: phylogenetic analysis by maximum likelihood. *Mol Biol Evol* 2007;**24**:1586–91.
17. Stamatakis A. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinforma Oxf Engl* 2014;**30**:1312–3.
18. Ronquist F, Teslenko M, van der Mark P, *et al*. MrBayes 3.2: efficient Bayesian phylogenetic inference and model choice across a large model space. *Syst Biol* 2012;**61**:539–42.
19. Boeckmann B, Bairoch A, Apweiler R, *et al*. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. *Nucleic Acids Res* 2003;**31**:365–70.
20. Ribeiro AJM, Holiday GL. Mechanism and catalytic site atlas (M-CSA): a database of enzyme reaction mechanisms and active sites. *Nucleic Acids Res* 2018;**46**:618–23.
21. Hon J, Borko S, Stourac J, *et al*. EnzymeMiner: automated mining of soluble enzymes with diverse structures, catalytic properties and stabilities. *Nucleic Acids Res* 2020;**48**:W104–9.
22. Altschul SF, Madden TL, Schaffer AA, *et al*. PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 1997;**17**:3389–402.
23. NCBI Resource Coordinators. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res* 2016;**44**:D7–19.
24. Camacho C, Coulouris G, Avagyan V, *et al*. BLAST+: architecture and applications. *BMC Bioinforma* 2009;**10**:421.
25. Edgar RC. Search and clustering orders of magnitude faster than BLAST. *Bioinforma Oxf Engl* 2010;**26**:2460–1.
26. Mirarab S, Nguyen N, Guo S, *et al*. PASTA: ultra-large multiple sequence alignment for nucleotide and amino-acid sequences. *J Comput Biol* 2015;**22**:377–86.
27. Katoh K, Standley DM. MAFFT multiple sequence alignment software version 7: improvements in performance and usability. *Mol Biol Evol* 2013;**30**:772–80.
28. Price MN, Dehal PS, Ap A. FastTree 2—approximately maximum-likelihood trees for large alignments. *PLoS One* 2010;**5**(3):e9490. doi: 10.1371/journal.pone.0009490.

29. Menardo F, Loiseau C, Brites D, *et al*. Treemmer: a tool to reduce large phylogenetic datasets with minimal loss of diversity. *BMC Bioinformatics* 2018;**19**:164.
30. Sievers F, Wilm A, Dineen D, *et al*. Fast, scalable generation of high-quality protein multiple sequence alignments using Clustal omega. *Mol Syst Biol* 2011;**7**:539.
31. Nguyen L-T, Schmidt HA, von Haeseler A, *et al*. IQ-TREE: a fast and effective stochastic algorithm for estimating maximum-likelihood phylogenies. *Mol Biol Evol* 2015;**32**:268–74.
32. Tria FDK, Landan G, Dagan T. Phylogenetic rooting using minimal ancestor deviation. *Nat Ecol Evol* 2017;**1**:193.
33. Hanson-Smith V, Kolaczkowski B, Thornton JW. Robustness of ancestral sequence reconstruction to phylogenetic uncertainty. *Mol Biol Evol* 2010;**27**:1988–99.
34. Sussman JL, Lin D, Jiang J, *et al*. Protein data Bank (PDB): database of three-dimensional structural information of biological macromolecules. *Acta Crystallogr D Biol Crystallogr* 1998;**54**:1078–84.
35. Biasini M, Schmidt T, Bienert S, *et al*. OpenStructure: an integrated software framework for computational structural biology. *Acta Crystallogr D Biol Crystallogr* 2013;**69**: 701–9.
36. Amin N, Liu AD, Ramer S, *et al*. Construction of stabilized proteins by combinatorial consensus mutagenesis. *Protein Eng Des Sel* 2004;**17**:787–93.
37. Lehmann M, Loch C, Middendorf A, *et al*. The consensus concept for thermostability engineering of proteins: further proof of concept. *Protein Eng Des Sel* 2002;**15**: 403–11.
38. Sullivan BJ, Nguyen T, Durani V, *et al*. Stabilizing proteins from sequence statistics: the interplay of conservation and correlation in triosephosphate isomerase stability. *J Mol Biol* 2012;**420**:384–99.
39. Shank SD, Weaver S, Kosakovsky Pond SL. Phylotree.Js—a JavaScript library for application development and interactive data visualization in phylogenetics. *BMC Bioinformatics* 2018;**19**:276.
40. Hanson RM, Prilusky J, Renjian Z, *et al*. JSmol and the next-generation web-based representation of 3D molecular structure as applied to Proteopedia. *Isr J Chem* 2013;**53**:207–16.
41. Maguire E, Rocca-Serra P, Sansone S-A, *et al*. Redesigning the sequence logo with glyph-based approaches to aid interpretation. In: *Proceedings of EuroVis 2014 Short Paper, IEEE Visualization and Graphics Technical Committee (IEEE VGTC)* 2014.
42. Kirmani S. A user friendly approach for design and economic analysis of standalone SPV system. *Smart Grid Renew Energy* 2015;**06**:67–74.
43. de Vienne DM, Aguileta G, Ollier S. Euclidean nature of phylogenetic distance matrices. *Syst Biol* 2011;**60**:826–32.
44. Bordewich M, Semple C. On the computational complexity of the rooted subtree prune and Regraft distance. *Ann Comb* 2005;**8**:409–23.
45. Harrower M, Brewer CA. ColorBrewer.Org: an online tool for selecting colour schemes for maps. *Cartogr J* 2003;**40**:27–37.
46. Ashkenazy H, Penn O, Doron-Faigenboim A, *et al*. FastML: a web server for probabilistic reconstruction of ancestral sequences. *Nucleic Acids Res* 2012;**40**:W580–4.
47. Diallo AB, Makarenkov V, Blanchette M. Ancestors 1.0: a web server for ancestral sequence reconstruction. *Bioinforma Oxf Engl* 2010;**26**:130–1.
48. Westesson O, Barquist L, Holmes I. HandAlign: Bayesian multiple sequence alignment, phylogeny and ancestral reconstruction. *Bioinforma Oxf Engl* 2012;**28**:1170–1.
49. Iwasaki I, Utsumi S, Ozawa T. New colorimetric determination of chloride using mercuric thiocyanate and ferric ion. *Bulletin of the Chemical Society of Japan* 1952;**25**(3):226.

FireProtDB: Database of Manually Curated Protein Stability Data.

Published online 9 November 2020 *Nucleic Acids Research, 2021, Vol. 49, Database issue* ***D319–D324***
doi: 10.1093/nar/gkaa981

FireProt^{DB}: database of manually curated protein stability data

Jan Stourac[1,2,†], Juraj Dubrava[1,3,†], Milos Musil[1,2,3], Jana Horackova[1], Jiri Damborsky[1,2], Stanislav Mazurenko[1,*] and David Bednar[1,2,*]

[1]Loschmidt Laboratories, Department of Experimental Biology and RECETOX, Masaryk University, Brno, Czech Republic, [2]International Clinical Research Center, St. Anne's University Hospital Brno, Brno, Czech Republic and [3]Department of Information Systems, Faculty of Information Technology, Brno University of Technology, Brno, Czech Republic

Received August 14, 2020; Revised September 18, 2020; Editorial Decision October 09, 2020; Accepted October 12, 2020

ABSTRACT

The majority of naturally occurring proteins have evolved to function under mild conditions inside the living organisms. One of the critical obstacles for the use of proteins in biotechnological applications is their insufficient stability at elevated temperatures or in the presence of salts. Since experimental screening for stabilizing mutations is typically laborious and expensive, *in silico* predictors are often used for narrowing down the mutational landscape. The recent advances in machine learning and artificial intelligence further facilitate the development of such computational tools. However, the accuracy of these predictors strongly depends on the quality and amount of data used for training and testing, which have often been reported as the current bottleneck of the approach. To address this problem, we present a novel database of experimental thermostability data for single-point mutants FireProt^{DB}. The database combines the published datasets, data extracted manually from the recent literature, and the data collected in our laboratory. Its user interface is designed to facilitate both types of the expected use: (i) the interactive explorations of individual entries on the level of a protein or mutation and (ii) the construction of highly customized and machine learning-friendly datasets using advanced searching and filtering. The database is freely available at https://loschmidt.chemi.muni.cz/fireprotdb.

INTRODUCTION

Proteins play essential roles in many biotechnological and biomedical applications, where they are often subjected to extreme environments, e.g. elevated temperatures or the presence of various salts. However, naturally occurring proteins have mostly evolved to function in the mild environmental conditions, and therefore their applicability is limited in the industrial applications. For this reason, protein engineers generally aim to improve protein stability, and thermostability is one of their primary targets (1) as it is correlated with serum survival time (2), half-life (3), expression yield (4) and activity in the presence of denaturants (5). A reliable assessment of the effect of a mutation on protein stability is often performed experimentally. Extensive experimental screening, however, is slow and costly, prompting the use of *in silico* approaches for the pre-selection of promising mutations. These methods are usually based on one of the three principles: (i) free energy calculations, (ii) phylogenetics or (iii) machine learning. With the recent advances in artificial intelligence, tool developers increasingly resort to the third group of methods. However, the accuracy of the machine learning-based predictors is still severely limited by the lack of high-quality data (6). Experimental characterizations are usually not capable of producing large amounts of data, and the majority of these measurements are scattered in the scientific literature. Thus, there is a strong demand for systematic collection, validation, and organization of such data in a database.

Two attempts have been made to establish a systematic and extensive collection of thermostability data so far. The first and largest database is the Thermodynamic Database for Proteins and Mutants–ProTherm (7). It was first released in 1999 with the aim to collect experimentally determined thermodynamic parameters for wild-type proteins

To whom correspondence should be addressed. Tel: +420 605 143 394; Email: davidbednar1208@gmail.com
Correspondence may also be addressed to S. Mazurenko. Email: mazurenko@mail.muni.cz
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.
Website address: https://loschmidt.chemi.muni.cz/fireprotdb.

and their mutants from the published literature. Its latest version contains >25 000 entries from 740 proteins, and it serves as the primary source of protein stability data for the development of new predictors. However, ProTherm was last updated in 2013 so the database is already out-of-date. Moreover, several critical issues have been reported, such as inaccurate annotations or wrong signs of values (6,8–10). This makes ProTherm even more difficult to use as time-demanding manual filtering and validation steps are required to confirm the values in the original articles. This manual filtering led to the construction of many different, often overlapping, subsets with corrected values and occasionally new data. Some of these derivative datasets were deposited to the VariBench database (11) without any attempts to reintegrate the changes into ProTherm or create an improved database. This changed in 2018 when ProtaBank (12) was released. This database aims to collect a wide range of protein engineering data such as thermostability, activity, expression, binding and several others. The developers imported all the data from ProTherm, yet they did not seem to perform any manual curation. Therefore, the critical issues listed above were not resolved. And while ProtaBank enriched the ProTherm data with recent experimental studies, the database does not offer any advanced searching and filtering capabilities, at least in its non-commercial version. This makes the data extraction and processing tedious by necessitating many manual steps and hindering the application of such data-driven methods as machine learning.

To overcome these limitations, we established the FireProt^{DB} database that holds manually curated thermostability data for single-point mutants. The database contains the data available in ProTherm, ProtaBank, and our extensive manual literature search. Its user-friendly interface allows easy and interactive browsing through the experimental data and provides links to the corresponding UniProt and PDB entries. Moreover, advanced searching and filtering capabilities, the ability to download the data in a simple table format, and meticulous labelling of data entries used for training and testing of published tools prompt the further application of machine learning.

MATERIALS AND METHODS

Database architecture and data model

The top-level entity of the FireProt^{DB} database is a unique protein sequence entry with the assigned UniProt ID (13). Protein sequences were preferred to structures due to the broader availability of the former. Each sequence is a string of amino acids in specified positions. Multiple mutations can be assigned to a single position, and each mutation can be evaluated by multiple measurements and derived values. The measurements represent the experimental values of the Gibbs free energy changes upon mutation ($\Delta\Delta G$) or changes in melting temperatures (ΔT_m). The derived values stand for averages or medians of multiple measurements for a particular mutation. Each measurement is also accompanied by a curation flag that indicates whether the value was manually validated against the original publication to guarantee its correctness. Furthermore, each measurement and derived value can be assigned to multiple published datasets to promote accurate validation and benchmarking of computational tools.

From the structural point of view, each sequence can have one or more assigned biological units that denote biologically relevant quaternary structures of asymmetric units stored in the PDB database (14). For representative biological units, the HotSpot Wizard 3.0 (15) calculation was executed to compute additional sequential and structural annotations. These annotations can help with the analysis of selected mutations and serve as pre-calculated features applicable in machine learning models.

Stability data acquisition and curation

FireProt^{DB} is composed of the data from four sources: the ProTherm database, the ProtaBank database, manual mining of the scientific literature, and data collected in our laboratory (Figure 1). The primary data source was ProTherm. Due to the multiple problems mentioned in the introduction, we followed several filtering steps. In the first step, we retained only those entries that met the following four criteria: (i) they have a single-point mutation; (ii) the mutation is not an insertion or deletion; (iii) the protein has a SwissProt accession code and/or a PDB identifier; (iv) the entry includes a measured $\Delta\Delta G$ and/or ΔT_m. Secondly, we performed a validity check of SwissProt accession codes and updated obsolete entries. ProTherm references mutations by their structure index, i.e., the residue number in the structure, which in many cases does not match their sequence index, i.e. the position in the sequence. To overcome this issue, we used a similar approach as in PDBSWS (16): use the Needleman-Wunsch algorithm (17) to construct the global sequence alignment of sequences extracted from PDB and UniProt entries and map the mutations onto the UniProt sequences. In the next step, we confirmed that the reported wild-type amino acids are in the correct positions in the structures and unified the reported units. Finally, we matched the data with the manually curated entries in the FireProt dataset (18), updated the values, and marked them as 'curated'.

In addition to ProTherm, we explored the studies reported in the ProtaBank database, extracted the thermostability data, and integrated them into our database. We also performed a manual literature search using stability-based keywords such as 'protein stability', 'thermostability', 'free energy upon mutation', 'protein stabilization'. We mined the recent scientific articles reporting mutants with measured stability data and contacted the authors of the publications when the relevant data were not available in the article. All such entries were marked as 'curated' as we extracted them directly from the original publications. Finally, we reviewed the thermostability data collected in our lab throughout the last few years and added them to the database. We perform experimental protein characterization in our protein engineering projects on a regular basis, and measuring protein stability is an essential part of such characterization. In total, the three sources led to a significant enlargement of the data size by 62% in terms of all the entries. The number of curated entries more than dou-

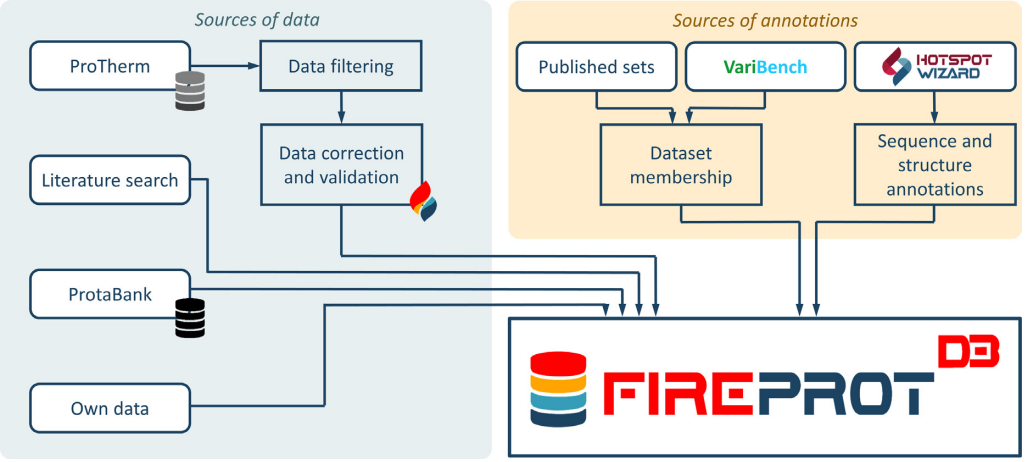

Figure 1. A schematic representation of the data comprising FireProt^{DB}. The primary source of data is filtered ProTherm (7). The FireProt data subset (18) was manually curated, compared to the source publications, and marked with the 'curated' flag. The publications from ProtaBank (12) and manual literature search were also used to deposit the data. Each mutation in the deposited data was annotated according to its membership in the published datasets and those deposited on VariBench (11). The HotSpot Wizard 3.0 (15) annotation tool was applied to each protein entry with a known tertiary structure.

bled compared to the previously collected cleaned FireProt subset of ProTherm.

Dataset assignment

In the second acquisition step, we collected 40 datasets from the VariBench database (11) and literature (18), which were used previously for training or testing of existing predictors. Since all these datasets are at least partially derived from ProTherm, we could label each measurement in FireProt^{DB} by its membership in the datasets. These labels are particularly useful for the comparison of new prediction models to the existing tools. This task is usually done by the performance evaluation of predictors on a dataset that is entirely independent of the training and test sets used for the development of the tools. Since the dataset construction is often laborious and consists of a manual data processing, the possibility to directly exclude the data present in given datasets significantly simplifies and speeds up the construction process.

Calculation of additional annotations

To provide our users with a more advanced description of their proteins of interest, we enriched the database by several important sequence- and structure-related information. These calculations were performed by HotSpot Wizard 3.0 (15), which is currently the only tool capable of deriving all these features in a single calculation (19) and provides machine-readable results. HotSpot Wizard was executed on a representative biological unit of each protein and provided the annotations for a structure, such as the residues located in protein pockets and tunnels, and a sequence, such as catalytic residues, evolutionary conservation scores, back-to-consensus mutations, and correlated pairs. These annotations can be helpful for a better understanding of structure-function relationships as well as for generating features for machine learning.

RESULTS

Web interface

The web interface was designed for both types of expected users—protein chemists and software developers. Protein chemists are often looking for the thermostability evidence for their protein of interest, and they will benefit from its interactivity and details pages with additional information. Machine learning experts and bioinformaticians will be more interested in advanced filtering capabilities facilitating the process of construction of highly customized datasets for the training or assessment of various predictors. The entry point to the database is the search form, which allows browsing in two major ways: (i) a simple full-text search for querying the database using protein name, UniProt accession codes, PDB identifiers, protein names, publications, authors or organisms and (ii) an advanced search allowing the users to construct complex rules based on the relational algebra and all available database fields. The latter is one of the key features of FireProt^{DB} as it facilitates the construction of highly customized datasets needed for the development of new predictors.

Once the user clicks on the 'Search' button, they are redirected to the page with the result table. This table contains a list of available experiments, their basic annotations, and measured values. The table is paginated to eliminate possible performance issues and allows further interactive filtering of displayed values. The user can then easily export the search results in the CSV format using the 'Export' button at the top or the bottom of the page.

Clicking on a mutation name leads to a page with a more detailed view, showing all the data entries and datasets that include the selected mutation. Clicking on a protein name leads to a page providing the basic information such as UniProt accession code, organism and Enzyme Commission number, as well as detailed annotation of secondary structure, catalytic sites, natural variants and amino acid charges derived from UniProt database using interactive

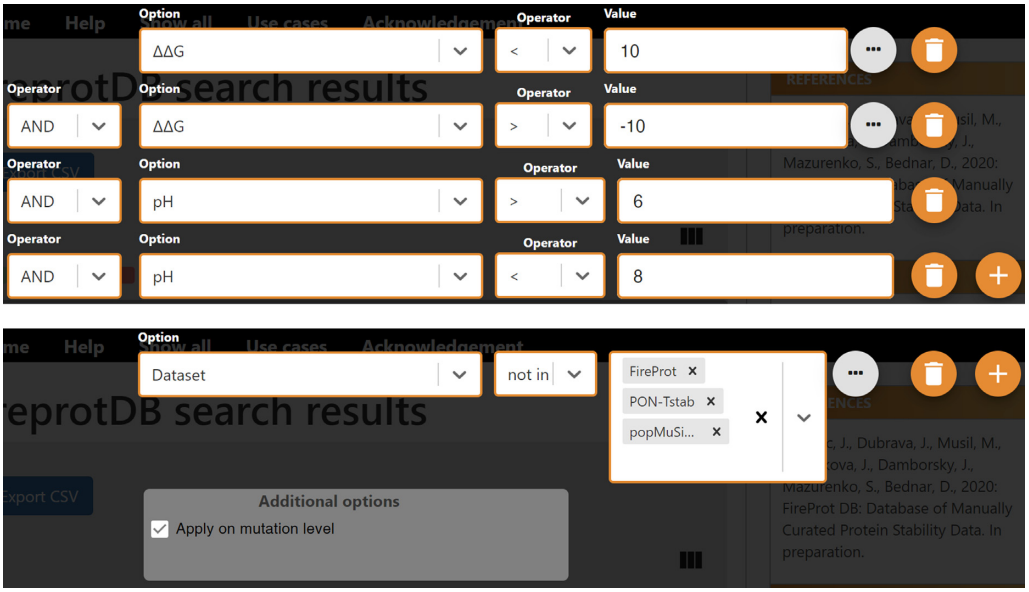

Figure 2. Examples of filtering protocols in FireProt^{DB}. **Top**: The request filters out the data collected at extreme pH or with extreme $\Delta\Delta G$ values, resulting in >3500 data points left. **Bottom**: An example of excluding all the mutations that appear in PopMuSiC, FireProt, or PON-Tstab datasets.

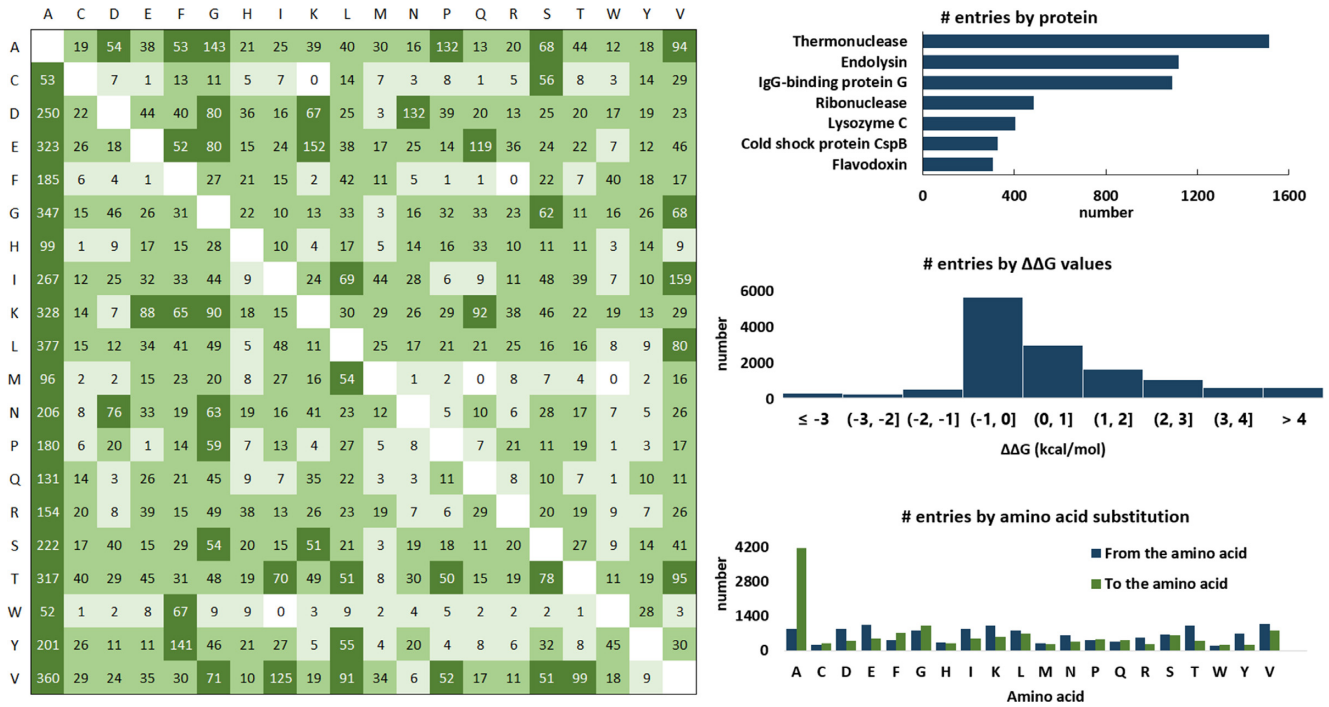

Figure 3. An overview of the data deposited to FireProt^{DB}. **Left**: The table shows the total number of each substitution pair with the wild type amino acids in rows, mutant amino acids in columns, and the coloring according to the thresholds of 1 (light green), 10 (medium green) and 50 (dark green) entries for the corresponding substitution. **Right**: Histograms showing the top seven proteins by their UniProt IDs, the $\Delta\Delta G$ values, and the cumulative number of amino acid substitutions.

ProtVista tracks (20). This page also contains a list of all known biological units and a table with all experimental measurements.

Search queries

Several types of search queries may be of interest to the users. The first one relates to data filtering by values (10). Typically, software developers filter out the data collected at extreme pH (<6 or >8) due to changes in charged states for ionizable residues. The entries with large absolute $\Delta\Delta G$ or ΔT_m are also sometimes excluded due to likely higher measurement errors, and also because dramatic changes to the stability may indicate significant structural alterations to the wild type, which may become a problem for structure-based features. The second type is relevant for benchmark-

ing of a newly designed predictor against the existing tools or creating a meta predictor. In either case, one usually needs to derive a data subset that has not been used by the existing predictors for training. The main reason is the robust performance estimate, which is typically over-optimistic for these sets (6). Two corresponding examples of such filtering protocols are shown in Figure 2.

Database dump

For the users requesting even higher control over the data and filtering capabilities, we offer the possibility to download the complete dump of the database in the SQL format. This data file can be easily imported to any modern MariaDB server, version 10.2, and higher. Since the database structure is complex and any custom query requires joining of multiple tables, the dump also contains a pre-defined view 'mutation_experiments_summary'. The summary combines all the tables and provides the data in a similar structure as the CSV export from the user interface. This view or its definition can serve as a useful starting point for additional filtering or creating custom queries.

Data statistics

Currently, FireProt^{DB} contains 13274 entries for 237 proteins (Figure 3), from which 8189 measurements originated from ProTherm. The remaining 5085 entries were added from our literature search (18%), publications from ProtaBank (28%), VariBench (53%), and our own records (1%). In total, 43% entries are destabilizing mutations ($\Delta T_m \leftarrow 1$ or $\Delta\Delta G > 1$ kcal/mol), 14% stabilizing ($\Delta T_m > 1$ or $\Delta\Delta G \leftarrow 1$ kcal/mol), and 43% considered neutral ($-1 \leq \Delta T_m \leq 1$ or $-1 \leq \Delta\Delta G \leq 1$ kcal/mol). The database also includes annotations for 40 various published datasets derived from ProTherm, deposited to VariBench (11), or available in the corresponding articles and web servers. As far as enzymes are concerned, those collected in the database cover the first six EC classes, three of which by >40% on the second level.

DISCUSSION

The availability of large high-quality datasets is one of the critical requirements for the advancement of machine learning-based *in silico* predictors. While some promising high-throughput experimental methods have been released recently (21,22), their validation is still ongoing, and protein stability experiments are still time-consuming and expensive. Building training and testing datasets is hindered by the data being hidden in the original articles, generating a strong demand for their systematic mining, collection, validation, and homogenization. The existing databases are not fulfilling all the requirements as ProTherm is outdated and contains incorrect data, and ProtaBank does not provide advanced search and export tools and is partly commercial.

FireProt^{DB} is a novel database for experimental thermostability data of protein single-point mutants. It consists of the data manually extracted from ProTherm, articles from ProtaBank, new data obtained by mining the recent literature, and the data collected in our laboratory. The database is accessible via a user-friendly graphical web interface allowing the users to search and browse the data interactively. Moreover, all the entries are annotated to indicate whether they belong to the already published datasets. These annotations, combined with the advanced searching and filtering capabilities, make FireProt^{DB} a valuable data resource for machine learning developers interested in constructing highly customized datasets.

In the future, we will improve our searching queries and employ automatic text-mining machine learning-based approaches (23–25) to accelerate literature mining and data collection, which will be followed by manual curation. We will also prepare an interactive form for data submissions by the users. Finally, we will extend the set of automatically generated features for mutations and add sequence similarity filtering to improve the data usability by the community of engineers applying machine learning to predict changes in protein stability.

FUNDING

Czech Ministry of Education, Youth and Sports [LQ1605, LM2015047, LM2018121, 02.1.01/0.0/0.0/18_046/0015975 to J.D.]; Operational Programme Research, Development and Education project MSCAfellow@MUNI [CZ.02.2.69 /0.0/0.0/17_050/0008496 to S.M.]; Brno University of Technology [FIT-S-20-6293 to M.M.]; CETOCOEN EXCELLENCE Teaming 2 project supported by Horizon2020 of the European Union [857560 to J.D.]; Czech Science Foundation [20-15915Y to D.B.]. Funding for open access charge: Czech ministry of Education, Youth and Sports [LM2015047].
Conflict of interest statement. None declared.

REFERENCES

1. Modarres,H.P., Mofrad,M.R. and Sanati-Nezhad,A. (2016) Protein thermostability engineering. *RSC Adv.*, **6**, 115252–115270.
2. Gao,D., Narasimhan,D.L., Macdonald,J., Brim,R., Ko,M.-C., Landry,D.W., Woods,J.H., Sunahara,R.K. and Zhan,C.-G. (2009) Thermostable variants of cocaine esterase for long-time protection against cocaine toxicity. *Mol. Pharmacol.*, **75**, 318–323.
3. Wijma,H.J., Floor,R.J. and Janssen,D.B. (2013) Structure- and sequence-analysis inspired engineering of proteins for enhanced thermostability. *Curr. Opin. Struct. Biol.*, **23**, 588–594.
4. Ferdjani,S., Ionita,M., Roy,B., Dion,M., Djeghaba,Z., Rabiller,C. and Tellier,C. (2011) Correlation between thermostability and stability of glycosidases in ionic liquid. *Biotechnol. Lett.*, **33**, 1215–1219.
5. Polizzi,K.M., Bommarius,A.S., Broering,J.M. and Chaparro-Riggers,J.F. (2007) Stability of biocatalysts. *Curr. Opin. Chem. Biol.*, **11**, 220–225.
6. Musil,M., Konegger,H., Hon,J., Bednar,D. and Damborsky,J. (2019) Computational design of stable and soluble biocatalysts. *ACS Catal.*, **9**, 1033–1054.
7. Kumar,M.D.S., Bava,K.A., Gromiha,M.M., Prabakaran,P., Kitajima,K., Uedaira,H. and Sarai,A. (2006) ProTherm and ProNIT: thermodynamic databases for proteins and protein–nucleic acid interactions. *Nucleic Acids Res.*, **34**, D204–D206.
8. Pucci,F., Bernaerts,K.V., Kwasigroch,J.M. and Rooman,M. (2018) Quantification of biases in predictions of protein stability changes upon mutations. *Bioinformatics*, **34**, 3659–3665.
9. Folkman,L., Stantic,B., Sattar,A. and Zhou,Y. (2016) EASE-MM: sequence-based prediction of mutation-induced stability changes with feature-based multiple models. *J. Mol. Biol.*, **428**, 1394–1405.

10. Mazurenko,S. (2020) Predicting protein stability and solubility changes upon mutations: data perspective. *Chem. Cat. Chem.*, **12**, doi:10.1002/cctc.202000933.
11. Sasidharan Nair,P. and Vihinen,M. (2013) VariBench: a benchmark database for variations. *Hum. Mutat.*, **34**, 42–49.
12. Wang,C.Y., Chang,P.M., Ary,M.L., Allen,B.D., Chica,R.A., Mayo,S.L. and Olafson,B.D. (2018) ProtaBank: a repository for protein design and engineering data. *Protein Sci.*, **27**, 1113–1124.
13. The UniProt Consortium (2019) UniProt: a worldwide hub of protein knowledge. *Nucleic Acids Res.*, **47**, D506–D515.
14. Jefferson,E.R., Walsh,T.P. and Barton,G.J. (2006) Biological units and their effect upon the properties and prediction of protein-protein interactions. *J. Mol. Biol.*, **364**, 1118–1129.
15. Sumbalova,L., Stourac,J., Martinek,T., Bednar,D. and Damborsky,J. (2018) HotSpot Wizard 3.0: web server for automated design of mutations and smart libraries based on sequence input information. *Nucleic Acids Res.*, **46**, W356–W362.
16. Martin,A.C.R. (2005) Mapping PDB chains to UniProtKB entries. *Bioinformatics*, **21**, 4297–4301.
17. Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.*, **48**, 443–453.
18. Musil,M., Stourac,J., Bendl,J., Brezovsky,J., Prokop,Z., Zendulka,J., Martinek,T., Bednar,D. and Damborsky,J. (2017) FireProt: web server for automated design of thermostable proteins. *Nucleic Acids Res.*, **45**, W393–W399.
19. Sequeiros-Borja,C.E., Surpeta,B. and Brezovsky,J. Recent advances in user-friendly computational tools to engineer protein function. *Brief. Bioinform.*, doi:10.1093/bib/bbaa150.
20. Watkins,X., Garcia,L.J., Pundir,S., Martin,M.J. and UniProt Consortium (2017) ProtVista: visualization of protein sequence annotations. *Bioinformatics*, **33**, 2040–2041.
21. Bunzel,H.A., Garrabou,X., Pott,M. and Hilvert,D. (2018) Speeding up enzyme discovery and engineering with ultrahigh-throughput methods. *Curr. Opin. Struct. Biol.*, **48**, 149–156.
22. Matreyek,K.A., Starita,L.M., Stephany,J.J., Martin,B., Chiasson,M.A., Gray,V.E., Kircher,M., Khechaduri,A., Dines,J.N., Hause,R.J. *et al.* (2018) Multiplex assessment of protein variant abundance by massively parallel sequencing. *Nat. Genet.*, **50**, 874–882.
23. Naderi,N. and Witte,R. (2012) Automated extraction and semantic analysis of mutation impacts from the biomedical literature. *BMC Genomics*, **13**, S10.
24. Witte,R. and Baker,C.J.O. (2007) Towards a systematic evaluation of protein mutation extraction systems. *J. Bioinform. Comput. Biol.*, **5**, 1339–1359.
25. Wei,C.-H., Harris,B.R., Kao,H.-Y. and Lu,Z. (2013) tmVar: a text mining approach for extracting sequence variants in biomedical literature. *Bioinformatics*, **29**, 1433–1439.

CAVER Analyst 2.0: Analysis and Visualization of Channels and Tunnels in Protein Structures and Molecular Dynamics Trajectories.

Bioinformatics, 34(20), 2018, 3586–3588
doi: 10.1093/bioinformatics/bty386
Advance Access Publication Date: 8 May 2018
Applications Note

OXFORD

Structural bioinformatics

CAVER Analyst 2.0: analysis and visualization of channels and tunnels in protein structures and molecular dynamics trajectories

Adam Jurcik[1], David Bednar[2,3], Jan Byska[4], Sergio M. Marques[2,3], Katarina Furmanova[1], Lukas Daniel[2,3], Piia Kokkonen[2,3], Jan Brezovsky[2,5,6], Ondrej Strnad[1], Jan Stourac[1,2,3], Antonin Pavelka[1,2], Martin Manak[7], Jiri Damborsky[2,3,*] and Barbora Kozlikova[1,*]

[1]Department of Computer Graphics and Design, Human Computer Interaction Laboratory, Faculty of Informatics, Masaryk University, 602 00 Brno, Czech Republic, [2]Loschmidt Laboratories, Department of Experimental Biology and Research Centre for Toxic Compounds in the Environment, Faculty of Science, Masaryk University, 625 00 Brno, Czech Republic, [3]International Centre for Clinical Research, St. Anne's University Hospital, 656 91 Brno, Czech Republic, [4]Visualization Group, Department of Informatics, University of Bergen, 5008 Bergen, Norway, [5]Department of Gene Expression, Institute of Molecular Biology and Biotechnology Faculty of Biology, Adam Mickiewicz University, 61-614 Poznan, Poland, [6]International Institute of Molecular and Cell Biology in Warsaw, 02-109 Warsaw, Poland and [7]NTIS – New Technologies for the Information Society, Faculty of Applied Sciences, University of West Bohemia, 301 00 Pilsen, Czech Republic

*To whom correspondence should be addressed.

Associate Editor: Alfonso Valencia

Received and revised on February 12, 2018; editorial decision on April 28, 2018; accepted on May 4, 2018

Abstract

Motivation: Studying the transport paths of ligands, solvents, or ions in transmembrane proteins and proteins with buried binding sites is fundamental to the understanding of their biological function. A detailed analysis of the structural features influencing the transport paths is also important for engineering proteins for biomedical and biotechnological applications.

Results: CAVER Analyst 2.0 is a software tool for quantitative analysis and real-time visualization of tunnels and channels in static and dynamic structures. This version provides the users with many new functions, including advanced techniques for intuitive visual inspection of the spatiotemporal behavior of tunnels and channels. Novel integrated algorithms allow an efficient analysis and data reduction in large protein structures and molecular dynamic simulations.

Availability and implementation: CAVER Analyst 2.0 is a multi-platform standalone Java-based application. Binaries and documentation are freely available at www.caver.cz.

Contact: kozlikova@fi.muni.cz or jiri@chemi.muni.cz

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The importance of access tunnels in proteins has been demonstrated by many studies in the last decade (Kingsley *et al.*, 2015; Marques *et al.*, 2017). Their examination in dynamical protein ensembles became a standard technique for studying important biochemical phenomena, designing new biocatalysts, materials or drugs (Brezovsky *et al.*, 2016; Gora *et al.*, 2013; Koudelakova *et al.*, 2013; Liskova *et al.*, 2015; Yu *et al.*, 2013). With the current computational capacity, it becomes affordable to obtain molecular dynamics (MD) trajectories up to the microsecond time scales. This trend requires new approaches to explore

the large datasets, as it becomes impracticable to observe such simulations in a frame-by-frame manner. Feature extraction and aggregation techniques, giving a guidance and overview of interesting sites and properties of tunnels over time, are therefore necessary. To follow this trend, we are introducing CAVER Analyst 2.0, which enables visual exploration of protein tunnels and channels even in microsecond-long MD simulations. This was achieved by introducing novel visualization approaches and other advanced functions, which enhance the manipulation of such simulation data. CAVER Analyst 2.0 introduces significant changes and improvements, focusing especially on large data processing, but also on providing the users with a complete description of the structural and biophysical features of protein tunnels and channels.

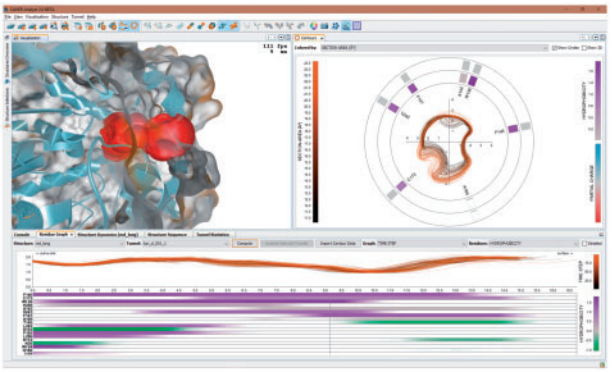

Fig. 1. CAVER Analyst 2.0 user interface

2 Features

Tunnel, channel and cavity calculation: CAVER Analyst 2.0 integrates the most up-to-date CAVER tool with the set of algorithms for: (i) identification of tunnels and channels in proteins, (ii) analysis of tunnels and channels in large MD simulations and (iii) identification of protein pockets and inner cavities. The algorithms are being continuously developed to provide the most accurate and computationally efficient description of these specific structural features. The tunnel calculation can be launched directly from the CAVER Analyst interface, which offers the basic and advanced calculation settings modes. For compatibility reasons, we keep the user interface of the Tunnel Computation window consistent with the version 1.0 (Kozlikova *et al.*, 2014). We have also improved the algorithm for the cavity detection (Manak *et al.*, 2017).

Visual analysis of tunnels: New visualization techniques present an important contribution to CAVER Analyst 2.0. They were mostly designed with the purpose of tunnel exploration in long MD trajectories (in AMBER, GROMACS, CHARMM formats), focusing on the changes of the tunnel properties and its surrounding residues over the time. Both techniques aggregate the spatial information to a single overview image so the user can get the information about the main trends in the tunnel behavior, regardless the MD simulation length. The first technique (Byska *et al.*, 2015) focuses on the visual representation of the shape of tunnel cross-cut at a specific site, e.g. its bottleneck. It shows its changes over time and physico-chemical properties of the amino acids lining that section (Fig. 1 and Supplementary Fig. S2). The central part is formed by the contour, which is defined by the cross-cut through a given tunnel. Each time step generates one contour and their overlay shows the shape of the cross-cut over the time. The rectangular bars surrounding the contours represent the respective lining amino acids colored by their physico-chemical properties. The second technique (Byska *et al.*, 2016) shows the width profile of a selected tunnel along the tunnel centerline (Fig. 1 and Supplementary Fig. S1). The amino acids forming the tunnel boundary are presented below the profile using a set of lines. The length of these lines illustrates the portion of the tunnel influenced by a particular amino acid. When dealing with dynamic ensembles, the lines represent the residues and their relative influence averaged over the entire simulation. Using a vertical slider, the user can specify a given section of the tunnel, for which the contour representation is calculated and visualized. The Supplementary Material demonstrates the applicability of these visualizations with two case studies focused on the engineering of tunnels aimed at improving protein stability and catalytic activity.

Mutagenesis: Engineering proteins typically requires the design and modeling of mutations. CAVER Analyst 2.0 supports this task by the new Mutagenesis Window (Supplementary Fig. S3). It offers the possibility to design one or more mutations in selected positions of a static molecule structure, which can be further used to recalculate the tunnels and visually compare the differences with the template. The newly designed molecule can be exported, upon which additional modeling studies, such as MD simulations, can be performed. The obtained trajectories can be loaded again to CAVER Analyst 2.0 and visually explored. The mutagenesis may use two different libraries of residue rotamers (Dunbrack *et al.*, 2011; http://bioserv.rpbs.univ-paris-diderot.fr/software.html).

Buffering: CAVER Analyst 2.0 enables to manipulate MD simulations of arbitrary length instantly, which ensures that the tool will be usable with simulations containing orders of magnitude higher number of time steps than now.

Other features: CAVER Analyst 2.0 offers advanced Measurement Window, the Clip Plane Window enabling to operate several independent clip planes and slices at once (Supplementary Fig. S4), improved manipulation of the protein structure, e.g. removing selected atoms, exporting structures from selected objects, video recording, high-resolution screenshots and the accessibility to common actions via the command line.

3 Implementation

CAVER Analyst 2.0 is a multi-platform JAVA-based software. It can run on both 32- and 64-bit system architectures with JAVA 1.8 (see Supplementary Material for implementation details).

Funding

Development of the software has been supported by the Czech Science Foundation (17-07690S and 16-06096S); by the Ministry of Education (LO1214, LO1506, LQ1605, LM2015047 and LM2015055); by PhysioIllustration research project 218023 funded by the Norwegian Research Council; and by National Science Centre, Poland 2017/25/B/NZ1/01307. Computational resources for simulations were supplied by the Ministry of Education, Youth and Sports of the Czech Republic under the Projects CESNET (LM2015042) and CERIT-Scientific Cloud (LM2015085).

Conflict of Interest: none declared.

References

Brezovsky,J. *et al.* (2016) Engineering a de Novo Transport Tunnel. *ACS Catalysis*, **6**, 7597–7610.

Byska,J. *et al.* (2015) MoleCollar and Tunnel Heat Map Visualization for Conveying Spatio-Temporo-Chemical Properties Across and Along Protein Voids. *Computer Graphics Forum*, **34**, 1–10.

Byska,J. *et al*. (2016) AnimoAminoMiner: exploration of Protein Tunnels and their Properties in Molecular Dynamics. *IEEE Transactions on Visualization and Computer Graphics*, **22**, 747–756.

Dunbrack,R.L. *et al*. (2011) A Smoothed Backbone-Dependent Rotamer Library for Proteins Derived from Adaptive Kernel Density Estimates and Regressions. *Structure*, **19**, 844–858.

Gora,A. *et al*. (2013) Gates of Enzymes. *Chemical Reviews*, **113**, 5871–5923.

Kingsley,L.J. *et al*. (2015) Substrate Tunnels in Enzymes: structure-Function Relationships and Computational Methodology. *Proteins*, **83**, 599–611.

Koudelakova,T. *et al*. (2013) Engineering Enzyme Stability and Resistance to an Organic Cosolvent by Modification of Residues in the Access Tunnel. *Angewandte Chemie International Edition*, **52**, 1959–1963.

Kozlikova,B. *et al*. (2014) CAVER Analyst 1.0: graphic Tool for Interactive Visualization and Analysis of Tunnels and Channels in Protein Structures. *Bioinformatics*, **30**, 2684–2685.

Liskova,V. *et al*. (2015) Balancing the Stability-Activity Trade-Off by Fine-Tuning Dehalogenase Access Tunnels. *ChemCatChem*, **7**, 648–659.

Manak,M. *et al*. (2017) Interactive Analysis of Connoly Surfaces for Various Probes. *Computer Graphics Forum*, **36**, 160–172.

Marques,S. *et al*. (2017) Enzyme Tunnels and Gates as Relevant Targets in Drug Design. *Medicinal Research Reviews*, **37**, 1095–1139.

Yu,X. *et al*. (2013) Conformational Diversity and Ligand Tunnels of Mammalian Cytochrome P450s. *Biotechnology and Applied Biochemistry*, **60**, 134–145.

CaverDock: A Molecular Docking-Based Tool to Analyse Ligand Transport through Protein Tunnels and Channels.

Bioinformatics, 2019, 1–8
doi: 10.1093/bioinformatics/btz386
Advance Access Publication Date: 11 May 2019
Original Paper

Structural bioinformatics

CaverDock: a molecular docking-based tool to analyse ligand transport through protein tunnels and channels

Ondrej Vavra[1,2,‡], Jiri Filipovic[3,‡], Jan Plhak[3], David Bednar[1,2], Sergio M. Marques[1,2], Jan Brezovsky[1,†], Jan Stourac[1,2], Ludek Matyska[3] and Jiri Damborsky[1,2,*]

[1]Loschmidt Laboratories, Department of Experimental Biology and RECETOX, Masaryk University, Brno 625 00, Czech Republic, [2]International Centre for Clinical Research, St. Anne's University Hospital Brno, Brno 656 91, Czech Republic and [3]Institute of Computer Science, Masaryk University, Brno 602 00, Czech Republic

*To whom correspondence should be addressed.
[†]Present address: Laboratory of Biomolecular Interactions and Transport, Department of Gene Expression, Institute of Molecular Biology and Biotechnology Faculty of Biology, Adam Mickiewicz University, Umultowska 89, 61-614 Poznan, Poland; International Institute of Molecular and Cell Biology in Warsaw, Ks Trojdena 4, 02-109, Warsaw, Poland
[‡]The authors wish it to be known that, in their opinion, the first two authors should be regarded as Joint First Authors.

Associate Editor: Yann Ponty

Received on December 31, 2018; revised on April 11, 2019; editorial decision on April 30, 2019; accepted on May 5, 2019

Abstract

Motivation: Protein tunnels and channels are key transport pathways that allow ligands to pass between proteins' external and internal environments. These functionally important structural features warrant detailed attention. It is difficult to study the ligand binding and unbinding processes experimentally, while molecular dynamics simulations can be time-consuming and computationally demanding.

Results: CaverDock is a new software tool for analysing the ligand passage through the biomolecules. The method uses the optimized docking algorithm of AutoDock Vina for ligand placement docking and implements a parallel heuristic algorithm to search the space of possible trajectories. The duration of the simulations takes from minutes to a few hours. Here we describe the implementation of the method and demonstrate CaverDock's usability by: (i) comparison of the results with other available tools, (ii) determination of the robustness with large ensembles of ligands and (iii) the analysis and comparison of the ligand trajectories in engineered tunnels. Thorough testing confirms that CaverDock is applicable for the fast analysis of ligand binding and unbinding in fundamental enzymology and protein engineering.

Availability and implementation: User guide and binaries for Ubuntu are freely available for non-commercial use at https://loschmidt.chemi.muni.cz/caverdock/. The web implementation is available at https://loschmidt.chemi.muni.cz/caverweb/. The source code is available upon request.

Contact: jiri@chemi.muni.cz

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Proteins are macromolecules that have myriads of functions in cells and uses in the chemical, biotechnological and pharmaceutical industries (Clouthier and Pelletier, 2012; Koeller and Wong, 2001; Soetaert and Vandamme, 2006). The majority of enzymes have their active site buried inside their core, connected with the external environment by access tunnels. Protein *tunnels* are characterized by a single opening. They enable the transport of substrates, products, solvent, and ions in and out of the active site (Brezovsky *et al.*, 2013; Gora *et al.*, 2013). Tunnels are essential for the natural function of enzymes, affecting their substrate specificity, stability and activity (Gora *et al.*, 2013). The shape and physicochemical properties of the tunnels may also protect proteins' hydrophobic core by restricting the access of solvent molecules and inhibitors. Protein *channels* are characterized by two openings. They are often connecting different cellular environments and play an essential role in the transport of various ligands, solvent molecules and ions. The rational modification of protein tunnels and channels is an important paradigm in protein engineering (Damborsky and Brezovsky, 2009). Tunnel- or channel-lining residues directly interact with the passing ligands and therefore represent hot spots for the optimization of various enzymatic properties (Bendl *et al.*, 2016).

The process of ligand transport cannot be studied easily by experimental techniques at the molecular level. Characterization of the transport processes is usually carried out indirectly by evaluation of enzymatic activity by steady state or transient kinetics (Biedermannova *et al.*, 2012; Hu *et al.*, 2016). Experimental methods offering a direct molecular description of the access pathways, like crystallography under xenon pressure (Milani *et al.*, 2005; de Sanctis *et al.*, 2004; Tilton *et al.*, 1984) or time-resolved protein crystallography (Schmidt *et al.*, 2005; Schotte *et al.*, 2003), are still very demanding and can only be applied to a narrow spectrum of proteins. Therefore, computational approaches provide an important insight into the molecular transport. Many of these methods involve perturbed molecular dynamics (MD) simulations (Arroyo-Mañez *et al.*, 2011) and other enhanced sampling methods (Rydzewski and Nowak, 2017). Methods such as Protein Energy Landscape Exploration (Borrelli *et al.*, 2005), Binding Free Energy Landscape (Bai *et al.*, 2013) or IterTunnel (Kingsley and Lill, 2014) were developed to simplify the setup and assessment of MD-based simulations. Nevertheless, MD-based methods are still difficult to use for interactive analyses, comparative studies or virtual screening campaigns due to the long simulation times and high numbers of repetitions required.

To analyse the ligand unbinding more rapidly, without the need for such computationally demanding MD methods, two alternative tools have been previously developed. SLITHER (Lee *et al.*, 2009) uses an iterative docking scheme to generate protein–ligand complexes and calculates corresponding binding free energies. This tool focuses on the study of ligands passing through channels inside a protein. The computational core of this method is molecular docking using AUTODOCK or MEDock (Chang *et al.*, 2005; Morris *et al.*, 2009). MoMA-LigPath (Devaurs *et al.*, 2013) uses a steric representation of molecules and a robotic Manhattan-like RRT algorithm (Cortes *et al.*, 2007) to explore the conformational space, but does not evaluate the free energy of the system. Therefore, an external method must be applied to quantify energy changes that occur during protein–ligand interactions along the tunnel. Here we present a novel method for simulating ligand binding and unbinding, implemented in the software tool CaverDock. The software is based on the step-wise movement of the ligand along the pre-calculated tunnel. CaverDock uses the docking algorithm of AutoDock Vina (Trott and Olson, 2010) enriched by the restraints, which serve to: (i) hold a selected atom of a ligand at a specific disc located along the tunnel or channel, i.e. position restraint; and (ii) dock the ligand in the upper-bound vicinity of a previous ligand conformation in order to maintain continuous ligand movement along the tunnel, i.e. pattern restraint.

2 Materials and Methods

2.1 CaverDock

In this section, we introduce the basic principles of CaverDock computation. A more thorough methodology is described in detail in Supplementary Material S1. The complete mathematical and algorithmic description of the method, which is beyond the scope of this study, is provided in Filipovic *et al.* (2019). The method is based on the step-wise movement of the ligand along the tunnel. The tunnel geometry, approximated by a sequence of spheres, is used as an input. This sequence of spheres can be obtained from tools providing the PDB file of the tunnel represented by spheres, such as CAVER 3.02 (Chovancova *et al.*, 2012), for whose output file format CaverDock was optimized. The sequence of spheres is then discretized into a sequence of discs (cross-section slices of a maximal thickness set by the user).

First, the selected ligand's atom is positioned at the disc by a position restraint. Second, CaverDock minimizes the ligand conformation and evaluates its binding free energy by using the scoring function from AutoDock Vina (Trott and Olson, 2010). Third, the ligand trajectory is produced by aggregating the docked poses of the ligand at each consecutive disc. Such a trajectory samples the tunnel thoroughly, but the movement of the ligand may be non-continuous. This non-continuous (lower-bound) trajectory is used to estimate the lower-bound (lowest) energy profile of the ligand's transport through the tunnel. The actual energy may be higher since the non-continuous movement can avoid small bottlenecks by rapid changes in the orientation or the conformation of the ligand.

Finally, the pattern constraint is used to compute the continuous (upper-bound) trajectory. In each step, the ligand is docked in the vicinity of its previous position allowing only small changes in the ligand conformation. The number of possible continuous trajectories grows exponentially with the number of discs, because each transition to a new disc may lead to changes in the ligand's position, orientation and conformation. Therefore, a heuristic method is employed to search for a continuous trajectory. When the binding free energy of a given docked conformation is significantly higher than the binding free energy of the conformation obtained from the lower-bound trajectory, backtracking is turned on. The ligand conformation is changed (e.g. to a conformation explored when lower-bound trajectory was computed) and the ligand is moved successively backward to previous discs. The backtracking ends when the forward and backward trajectories converge, or it is stopped if the starting disc is reached. As there is no guarantee that the resulting continuous trajectory is optimal, we call it the upper-bound trajectory as the actual energy may be lower than the computed energy.

The practical differences between lower-bound and upper-bound trajectories are the following: the lower-bound trajectory is able to completely sample the ligand trajectory. The information from the lower bound is sufficient for comparison purposes but its main limitation is that it can miss small bottlenecks by rapid changes of the ligand orientation. However, the sudden changes in orientation could potentially mimic the natural flexibility of the protein and lower the

unnatural energy barriers caused by the receptor rigidity during binding or unbinding. On the other hand, the upper-bound trajectory is completely smooth. However, it can create unrealistic conformations in very tight parts of the tunnel, which are signified by sudden sharp peaks in the binding energy profile. The energy profile from the lower-bound calculation shows the best-case scenario of the binding energy along the tunnel, while the upper-bound can report exaggerated energies because the respective trajectory may not be optimal.

2.2 Input preparation

With one exception (for Dataset III, Section 2.3.3), all the simulations described in this manuscript were performed using the following settings: the PDB files of the proteins were obtained from the RCSB Protein Data Bank (Berman *et al.*, 2000). MOL2 files of the ligands were either downloaded from the ZINC database (Irwin and Shoichet, 2005) or built in Avogadro (Hanwell *et al.*, 2012) and minimized with the MMFF94 force field (Halgren, 1996). The receptor and ligand PDBQT files were prepared using scripts from MGLtools (Morris *et al.*, 2009) with the default parameters. The tunnel calculation was performed by CAVER 3.02 (Chovancova *et al.*, 2012), with the size of the probe set to 0.7 Å, and the other parameters at the default values. The tunnels were discretized with 0.3 Å steps and extended by 2 Å in the direction of the vector calculated from the last two spheres in the original tunnel. The script for the tunnel extension is provided in the CaverDock package. The configuration files and calculation of the grid box containing the whole tunnel geometry were prepared by the provided preparation script. The dragged atom, i.e. the atom attracted to the middle of the disc at the beginning of each calculation step, was chosen using the default auto-selection (the closest atom to the centroid of the molecule). All the simulations were performed in the default (unbinding) direction. To simulate the binding process, the user has to invert the discretized tunnel file, e.g. using the bash command *tac*. Side-chain flexibility of selected residues can be prepared using MGLtools. Detailed information about the CaverDock setup is provided in the manual available at https://loschmidt.chemi.muni.cz/caverdock/.

2.3 Testing datasets

All datasets (Supplementary Material S20) together with the figures of the ligand's geometries (Supplementary Material S19) are provided as the Supplementary Material.

2.3.1 Dataset I: benchmarking

CaverDock was compared with the two existing tools for prediction of the ligand passage SLITHER (Lee *et al.*, 2009) and MoMA-LigPath (Devaurs *et al.*, 2013). These tools were compared using 10 cases. The dataset consists of six example cases presented at the websites of SLITHER and MoMA-LigPath, complemented with other systems found in the literature (Cui *et al.*, 2015; Koudelakova *et al.*, 2011; Peräkylä, 2009; Wang *et al.*, 2005) (Supplementary Material S2). SLITHER and MoMA-LigPath are available as web servers. SLITHER calculations were conducted with the AUTODOCK algorithm and the default rigid receptor. MoMA-LigPath was used with the default settings. The side chains are treated as flexible by default only in the case of MoMA-LigPath. CaverDock was used with a rigid receptor and the calculations were set up in the same manner as described in the Section 2.2.

2.3.2 Dataset II: geometry of tunnels

This dataset was used to test the ability of CaverDock to model ligand trajectories through tunnels with a broad range of geometries. The data for proteins and their corresponding tunnels were collected from the literature (Chovancova *et al.*, 2012; Koudelakova *et al.*, 2013). Information about the proteins' native substrates was obtained from the UniProt (The UniProt Consortium, 2017) and BRENDA databases (Schomburg *et al.*, 2004). The complete dataset consists of 26 proteins with 113 identified tunnels and 33 natural substrates, creating altogether 136 cases (Supplementary Material S3).

2.3.3 Dataset III: geometry of substrates

The correspondence between the binding energies from CaverDock and experimentally measured kinetics data were validated using this dataset. The haloalkane dehalogenase LinB (PDB ID: 1K63) and the set of 25 halogenated substrates with experimentally determined K_M values (Kmunícek *et al.*, 2005) were used (Supplementary Material S4). To ensure the complete unbinding of each ligand, we used specific settings to calculate the tunnels in CAVER 3.02, with the shell radius and shell depth set to 20 and 4 Å, respectively. We selected the tunnels corresponding to the p1 and p2 tunnels of the LinB dehalogenase and extended them by 20 Å.

2.3.4 Dataset IV: tunnel engineering

This dataset was assembled to test the ability of CaverDock to describe the differences in enzymes with rationally engineered access tunnels (Supplementary Material S5). We analysed the wild-type dehalogenase LinBWT (PDB ID: 1K63), the variant LinB32 (PDB ID: 4WDQ) with closed main p1 tunnel (LinB-closedW), and the variant LinB86 (PDB ID: 5LKA) with newly open p3 tunnel (LinB-open^{W}). The goal was to compare the energy profiles from CaverDock (i) with the tunnels detected in the crystal structures and (ii) the frequency of product (2-bromoethan-1-ol) release through p1 and p3 tunnels obtained in previously published MD simulations (Brezovsky *et al.*, 2016).

3 Results

3.1 Illustration of CaverDock output

CaverDock generates an output in the form of two PDBQT files. One file provides a smoothed upper-bound trajectory while the other represents the lower-bound trajectory of the ligand. Information about the binding energies and tunnel radii is listed in the REMARK lines of the respective ligand trajectories, and can be extracted and plotted using the scripts provided with the package. The visualization of the results obtained using the CaverDock is presented in Figure 1.

3.2 Comparison of CaverDock with state-of-the-art methods SLITHER and MoMA-LigPath

We studied the robustness of SLITHER, MoMA-LigPath and CaverDock using the Dataset I (Table 1). SLITHER was able to predict the unbinding trajectory for half of the tested systems. Its main limitation is that the ligands are moved in a direction parallel to the *y*-axis only, making the analysis of curved and narrow tunnels difficult. Moreover, the ligand trajectories calculated by SLITHER are discontinuous with significant gaps between the predicted ligand positions. MoMA-LigPath was more successful, providing a continuous trajectory for 6 of the 10 test cases, but the tool does not provide any energy information. CaverDock was the most robust and provided results for all 10 test cases.

A critical comparison of the features of individual tools revealed that the main advantage of CaverDock over SLITHER is its ability to calculate the ligand transport in any direction with a simple setup

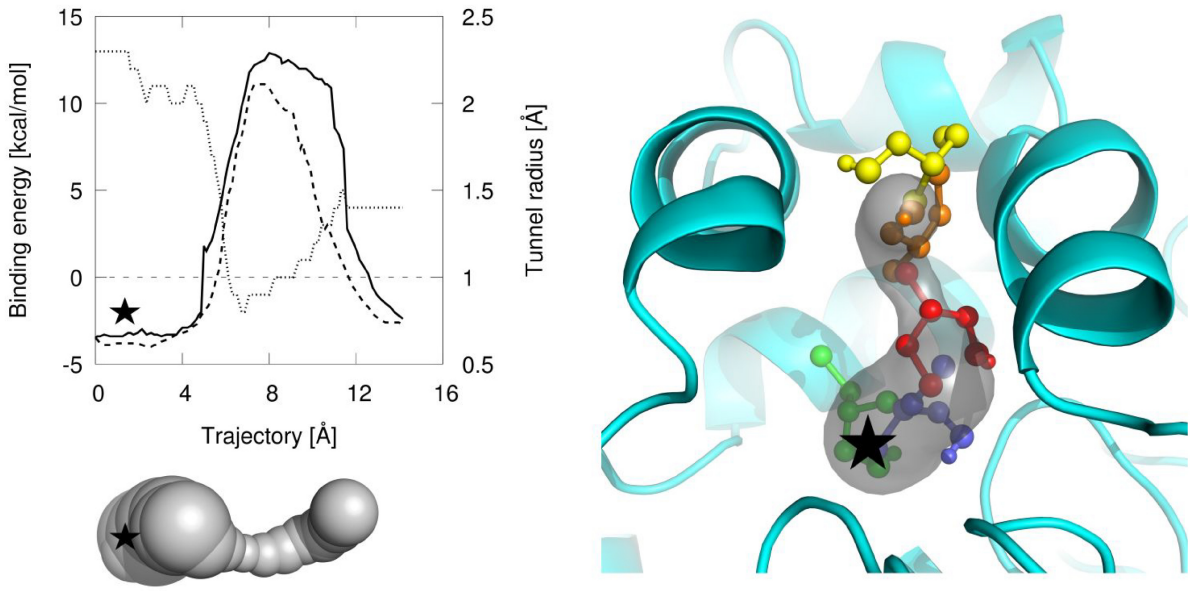

Fig. 1. Illustration of the results obtained using the CaverDock. Top left: Examples of the energy profiles for the haloalkane dehalogenase LinB (PDB ID: 1K63) and 2,3-dichloropropan-1-ol. The binding energy (left vertical axis) of a smoothed continuous upper-bound trajectory, the lower-bound trajectory and tunnel radius (right vertical axis) are indicated by the full line, dashed line and dotted line, respectively. The direction of the trajectory in the plot is from the active site (marked by the star symbol) to the surface of the protein. Bottom left: The three-dimensional surface of the corresponding tunnel calculated by CAVER 3.02. Right: Visualization of a part of a CaverDock trajectory. The protein is displayed as the cyan cartoon with the tunnel shown as the grey transparent surface. Selected snapshots of the ligand are shown in ball-and-stick representation: 1 (green), 10 (blue), 45 (red), 60 (orange) and 85 (yellow). The snapshot 45 (red) corresponds to the binding energy maximum of the energy profile

Table 1. Comparative study of CaverDock, SLITHER and MoMA-LigPath

PDB ID	Protein	Ligand	Ligand passage		
			CaverDock	SLITHER	MOMA-LigPath
1BN7	Haloalkane dehalogenase	1-Chlorobutane	Yes	Yes	Yes
1MAH	Acetylcholinesterase	Acetylcholine	Yes	Yes	Yes
2A65	Leucine transporter	Leucine	Yes	Yes	Yes
1PV7	Lactose permease	Lactose	Yes	Yes	Yes
1SUK	Glucose transporter	α-D-Glucopyranose	Yes	Yes	No
1TCC	Lipase B	4-Methyloctanoic acid	Yes	No	Yes
1ZNJ	Insulin hexamer	Phenol	Yes	No	Yes
1RC2	Aquaporin Z	Glycerol	Yes	No	No
1IE9	Vitamin D receptor	1, 25-Dihydroxyvitamin D3	Yes	No	No
3LC4	Cytochrome P450 2E1	Arachidonic acid	Yes	No	No

Note: Yes and No describes the result of the qualitative test whether the tool was able to predict a ligand's trajectory.

(Supplementary Material S6). The resolution of CaverDock trajectories is much higher since the ligand has to move through each disc of the discretized tunnel or channel so there are no large gaps in the trajectory. On the other hand, CaverDock currently cannot analyse multiple protein conformations simultaneously as it is possible with the relaxed receptor mode of SLITHER. The main differences between CaverDock and MoMA-LigPath are that CaverDock is able to simulate also the binding trajectory of a ligand and gives the information about the binding energy along the pathway. The advantage of MoMA-LigPath is that it can treat the flexibility of many side chains simultaneously, while implementation of the side-chain flexibility is still rather limited in CaverDock and SLITHER. Finally, CaverDock is the only software which is provided as a web application as well as a standalone tool, making it suitable for extensive virtual screening campaigns. The features and the setup options of the tested tools are summarized in Supplementary Material S6.

3.3 Impact of tunnel geometries on CaverDock calculations

Dataset II was constructed to test the predictive power of CaverDock with proteins possessing various geometries of tunnels with their native substrates. CaverDock was tested on 26 proteins with 33 substrates (some proteins had more than one native substrate) and 113 tunnels, 136 calculations altogether. Out of 136 CaverDock runs, 81 finished with lower-bound and upper-bound trajectories, 44 finished only with lower-bound and in 11 cases the ligands were not able to pass through the tunnels (Supplementary Material S7).

Although a smoothed (upper-bound) trajectory was not calculated for almost half of the cases, this does not mean that CaverDock could not properly simulate the ligand unbinding. The ligand unbinding process was still sufficiently sampled along the whole tunnel in the lower-bound trajectory, although the transition

from one conformation to the next was not smooth. This assertion is corroborated by the manifestation of increases in energy caused by the tunnel bottlenecks in the lower-bound energy profiles alone. Therefore, providing data for the lower-bound trajectory alone is a valid result. Further analysis of the 11 cases in which the ligands could not pass through the tunnels in CaverDock simulations revealed that the failure was due to the tunnels being too narrow for the ligands. In all except one case, these tunnels were graded by CAVER 3.02 as being 'lower throughput', meaning they are apparently less important than others for transport and thus unlikely to be functionally relevant for transport of the respective ligands. Plots of the energy profiles can be found in Supplementary Material S8.

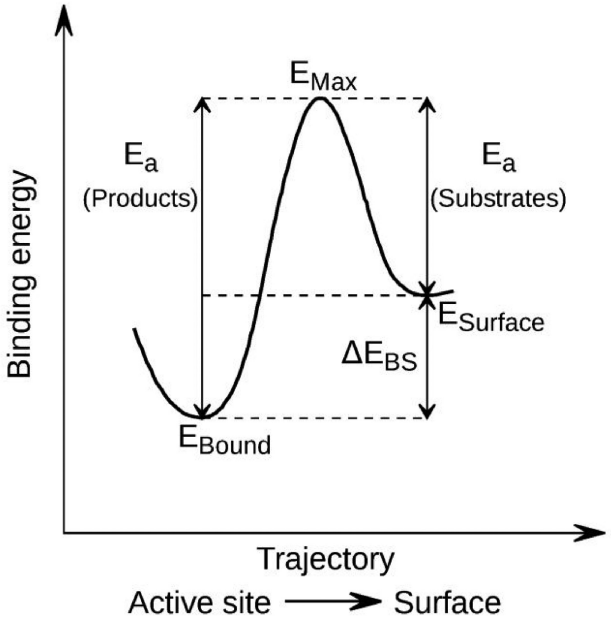

Fig. 2. Schematic energy profile with marked energy values. E_{Bound}, the binding energy of a ligand located inside the active site; E_{Max}, the highest binding energy in the trajectory; $E_{Surface}$, the binding energy of the ligand located at the protein surface; E_a, the activation energy of association for the products ($E_{Max} - E_{Bound}$) and for the reactants ($E_{Max} - E_{Surface}$), corresponds to the kinetics of a ligand passing through the tunnel; ΔE_{BS}, difference of the binding energies in the bound state and at the surface corresponds to the enthalpy of binding, and is related to the equilibrium constant

3.4 Validation of CaverDock calculations against experimental data

CaverDock was used to analyse the p1 and p2 tunnels of the haloalkane dehalogenase LinB with the set of halogenated substrates from Dataset III, for which the values of Michaelis constants have been determined experimentally in our laboratory (Kmunícek *et al.*, 2005). The impact of the ligand and tunnel geometry on the energy profiles was studied. Selected energy values (Fig. 2) were extracted from the energy profiles: (i) the energy minimum close to the start of the trajectory corresponding to the ligand bound into the active site (E_{Bound}), (ii) the maximum energy from the profile (E_{Max}) and (iii) the last minimum related with the surface-bound ligand ($E_{Surface}$). For the analysis of substrates, the main focus was devoted to the evaluation of the height of the activation energy of association (E_a) calculated for the ligands going through the tunnel into the active site. Therefore, the activation energy of association was calculated as $E_a = E_{Max} - E_{Surface}$. E_a can be related to the binding kinetics by the Arrhenius law ($k = Ae^{-\frac{E_a}{RT}}$), and thus E_a is expected to vary linearly with the logarithm of the association rate, k_{on}. The energy difference between the active site and the surface-bound energy (ΔE_{BS}) was calculated as the difference between the corresponding minima. ΔE_{BS} quantifies the enthalpy of binding, which is, according to the van't Hoff equation, negatively correlated with the logarithm of the equilibrium constant. Even though CaverDock provided the smoothed trajectories for all the test cases, we analysed the binding energies from the lower-bound trajectories, which provide more reasonable profiles.

Comparison of E_a values for the two tunnels (Fig. 3) indicates that the energy barriers for the ligand passage through the p2 tunnel

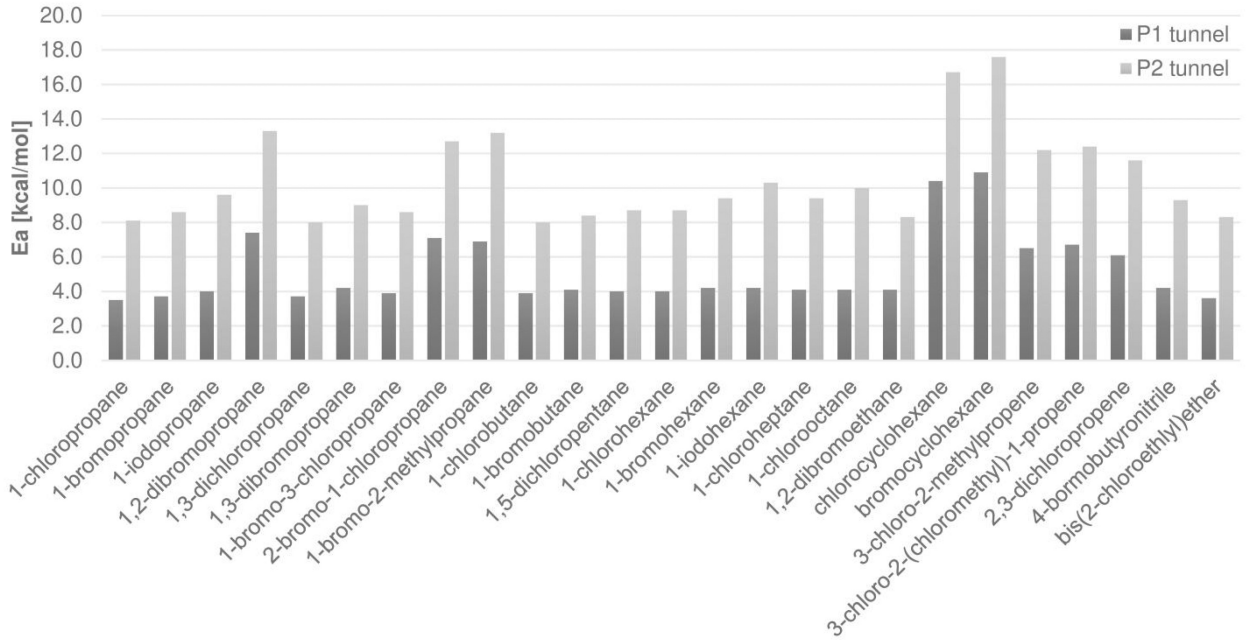

Fig. 3. Comparison of the activation energy of association E_a for the p1 and p2 tunnels of the haloalkane dehalogenase LinB with the set of 25 substrates

are typically two times higher than the corresponding barriers for the p1 tunnel (Supplementary Material S9 and S10). This is likely due to the fact that p2 is narrower, longer and more curved than p1 (Supplementary Material S11). These results suggest the preference of the substrates for binding into the buried active site of that protein using the p1 tunnel, which is the main tunnel observed in the crystal structures. Without exception, the energy minima of the bound states were lower than the surface-bound minima, showing the tendency of all substrates to bind into the active site cavity rather than any other part of the tunnel or surface.

We calculated the correlations between the analysed energy values and the experimentally measured Michaelis constants (K_M) and catalytic constants (k_{cat}). With crude approximation, K_M values should be related to the binding affinity, and it may be expected to correlate with ΔE_{BS}. The interpretation of K_M in various systems is complex, as it is composed of multiple steps in the enzymatic reaction, and not only composed of the binding process. Pearson's correlation coefficient of $\log(K_M)$ with ΔE_{BS} showed the values of 0.6 and 0.7 for the p1 and p2 tunnel, respectively. This statistically significant correlation shows that CaverDock can describe the binding trajectory and find a proper binding mode. The level of the observed correlation is in agreement with the nature of Michaelis constant for the haloalkane dehalogenase LinB, which is defined by a combination of the substrate binding and the rate of the follow-up S_N2 reaction step resulting in the covalently bound intermediate (Prokop *et al.*, 2003).

Regarding k_{cat}, this kinetic parameter is limited by the slowest step in the catalytic cycle. In the haloalkane dehalogenases, this cycle is rather complex, and the rate-limiting step can easily vary from substrate to a substrate (Prokop *et al.*, 2003). Therefore, k_{cat} is expected to correlate with the E_a barriers *only* in the systems where the binding of a substrate or unbinding of a product is the rate-limiting step of the catalysis. Pearson's correlation coefficient of k_{cat} and E_a is −0.2 for both p1 and p2 tunnels. These statistically insignificant correlations are in agreement with the transient kinetic analysis of the haloalkane dehalogenase LinB (Prokop *et al.*, 2003), demonstrating that the substrate binding is not the rate-limiting step in the catalytic cycle. The linear regressions of these correlations are provided in Supplementary Material S12.

3.5 Analysis of proteins with computationally designed access tunnels

We used CaverDock to analyse Dataset IV, the unbinding of the 2-bromoethan-1-ol product from three different LinB variants. The lower-bound energy profiles for the p1 and the p3 tunnels in LinB wild-type (LinBWT) and two variants carrying tunnel mutations LinB32 (LinB-closed^W) and LinB86 (LinB-open^W) are shown in Figure 4. The results from CaverDock calculation correspond well with the properties of the tunnels found in the crystal structures, supporting the blockage of the main p1 tunnel by the bulky Trp residue, intentionally introduced to the LinB32 and LinB86 variants (Brezovsky *et al.*, 2016). The narrowing of the p1 tunnel by this engineering step resulted in an increased energy barrier for the transport of the 2-bromoethan-1-ol from the active site to protein surface (Fig. 4A). The follow-up step of the project was opening *de novo* p3 tunnel in the protein LinB86. Calculation of the energy barriers for the release of 2-bromoethan-1-ol by this route clearly illustrates removal of the first barrier and significant lowering of the second barrier of the energetic profile (Fig. 4B), which is again in perfect agreement with crystallographic data. The calculated energy barriers are matching the diameters of p1 and p3 tunnels calculated in each

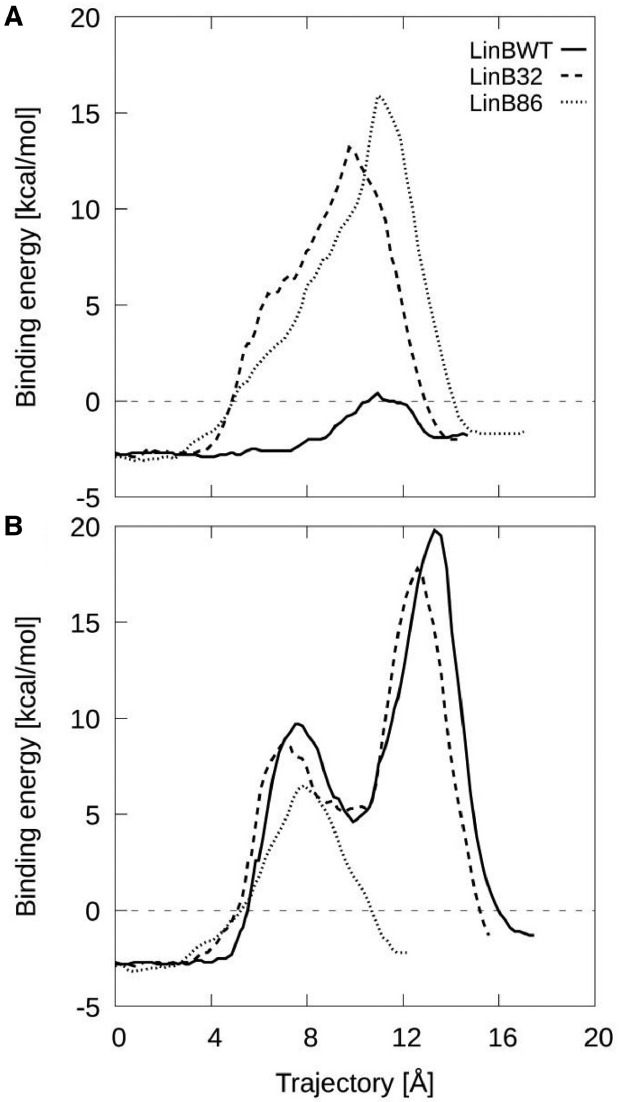

Fig. 4. Analysis of 2-bromoethan-1-ol unbinding through p1 (**A**) and p3 (**B**) tunnels in the LinB variants with rationally engineered tunnels (Brezovsky *et al.*, 2016). The p1 tunnel is blocked by a bulky Trp residue in LinB32 and LinB86, resulting in an increase in the energy barrier. The p3 tunnel was opened in LinB86 by three point mutations, resulting in removal of the first barrier and lowering the second barrier

of the experimental structures of LinBWT, LinB32 and LinB86 (Supplementary Material S13), suggesting that CaverDock calculations can reproduce tunnel engineering exercises and has a great potential for computational protein design targeting protein tunnels and channels.

4 Discussion

Tunnels and channels facilitate the transport of ligands through diverse proteins, so understanding the processes underlying the ligand transport is a cornerstone of biochemistry, structural biology and medicinal chemistry. The characteristics of these transport pathways are difficult to study using the currently available experimental techniques, which are not trivial to set up and are very time-consuming (Mittermaier and Meneses, 2013; Schotte *et al.*, 2003). Moreover, it can be difficult to study them with the currently available

computational tools, as they typically involve the use of MD simulations (Barducci *et al.*, 2010; Grubmüller *et al.*, 1996; Lüdemann *et al.*, 2000), which require substantial knowledge of the methods and too extensive computational resources for screening large number of ligands.

These limitations led us to develop CaverDock, a fast computational tool based on molecular docking for simulating ligand transition through protein tunnels and channels. CaverDock can be used to infer whether the studied ligand will likely pass through a particular protein tunnel. It can evaluate the passage of different ligands, or semi-quantitatively compare the difficulty of ligands passing through several different tunnels. The method is very easy to setup with the calculation times typically in the order of minutes. This makes it suitable for virtual screening purposes or for the enrichment of widely used virtual screening results by molecular docking (Daniel *et al.*, 2015). In comparison with MD simulations, CaverDock does not require extensive knowledge of the studied system. CaverDock is able to sample the binding energy throughout the whole protein tunnel and identify unfavourable binding interactions, which can then be optimized by site-directed mutagenesis (Kaushik *et al.*, 2018; Liskova *et al.*, 2017). Such places would be missed by traditional docking techniques. The easy setup and execution of the calculations may easily provide trajectories of the a ligand passage through a protein of interest, which can be used as educational materials in the biochemistry courses, assisting teachers with visualization of the process of ligand binding or unbinding. Finally, the advanced settings in CaverDock also enable constrained and pattern docking calculations.

The comparison presented here showed that, when aiming at exploring the properties of the ligand transport through molecular tunnels, CaverDock displayed better performance than the other tested tools SLITHER (Lee *et al.*, 2009) and MoMA-LigPath (Devaurs *et al.*, 2013). We thoroughly tested CaverDock using 69 ligands, 130 tunnel geometries and 40 protein structures. In most cases, the ligands successfully passed through the tunnels. In some cases, the steric hindrances prevented the calculation of a smoothed (upper-bound) continuous ligand trajectory. However, in these cases CaverDock was still able to calculate the non-continuous lower-bound trajectory. Further analysis of the lower-bound energy profiles showed that they reflect the increases in the energy associated with the ligands' passage through a more restricted and narrow spaces in a tunnel. Thus, the lower-bound trajectory alone is suitable for sampling all the binding energies through a tunnel. CaverDock's ability to calculate smoothed upper-bound trajectories could potentially be improved by choosing a dragged atom close to the edge of the ligand rather than the default atom closest to its centroid (especially for large ligands with high degrees of freedom). Another problem that may occur is that the energy at the end of the simulation (at the tunnel mouth) sometimes did not converge to zero. This implies that the ligand did not reach the fully unbound state. To ensure further unbinding of the ligand, the tunnel geometry obtained from CAVER may be prolonged or recalculated with different settings.

The most important limitation of the first version of CaverDock is that it cannot robustly address conformational dynamics of the protein structure. We have analysed the current implementation of flexibility of the sidechains (Trott and Olson, 2010) in CaverDock (Supplementary Material S14). The application of flexibility brought an overall lowering of the energy profile but at the same time it produced unlikely high-energy conformations of the protein structure in some instances (Supplementary Material S15, Supplementary Material S16). Moreover, the introduction of multiple side-chain flexibility significantly increased the calculation time. For now, we advise users to use the current implementation of sidechain flexibility cautiously and take practical measures such as minimizing the number of flexible sidechains and checking the generated protein conformations for steric clashes. We also looked at the importance of backbone dynamics (Supplementary Material S17). Using the snapshots from previously published accelerated molecular dynamics simulations, we have observed expected changes in the CaverDock energy profiles calculated with the structures of proteins possessing different conformations (Supplementary Material S18). These structures represent highly valuable benchmark for the rigorous treatment of protein flexibility, which is currently under development. The most important part in the development will be to balance the trade-off between the systematic description of protein conformations and the speed of CaverDock calculations.

Acknowledgements

The authors thank Antonin Pavelka (Masaryk University) for stimulating discussions during the initial phase of the project.

Funding

For financial support, the authors thank the Czech Ministry of Education [LM2015047, LM2015055, LM2015042, LM2015085, CZ.02.1.01/0.0/0.0/16_026/0008451, CZ.02.1.01/0.0/0.0/16_019/0000868, CZ.02.1.01/0.0/0.0/16_013/0001761] and European Commission [720776, 722610 and 814418]. Computational resources were provided by the CESNET and CERIT Scientific Cloud (LM2015042 and LM2015085). O.V. is the recipient of a Ph.D. Talent award provided by Brno City Municipality.

Conflict of Interest: none declared.

References

Arroyo-Mañez,P. *et al.* (2011) Protein dynamics and ligand migration interplay as studied by computer simulation. *Biochim. Biophys. Acta*, **1814**, 1054–1064.

Bai,F. *et al.* (2013) Free energy landscape for the binding process of Huperzine A to acetylcholinesterase. *Proc. Natl. Acad. Sci. USA*, **110**, 4273–4278.

Barducci,A. *et al.* (2010) Linking well-tempered metadynamics simulations with experiments. *Biophys. J.*, **98**, L44–6.

Bendl,J. *et al.* (2016) HotSpot Wizard 2.0: automated design of site-specific mutations and smart libraries in protein engineering. *Nucleic Acids Res.*, **44**, W479–W487.

Berman,H.M. *et al.* (2000) The Protein Data Bank. *Nucleic Acids Res.*, **28**, 235–242.

Biedermannova,L. *et al.* (2012) A single mutation in a tunnel to the active site changes the mechanism and kinetics of product release in haloalkane dehalogenase LinB. *J. Biol. Chem.*, **287**, 29062–29074.

Borrelli,K.W. *et al.* (2005) PELE: protein energy landscape exploration. A novel Monte Carlo based technique. *J. Chem. Theory Comput.*, **1**, 1304–1311.

Brezovsky,J. *et al.* (2013) Software tools for identification, visualization and analysis of protein tunnels and channels. *Biotechnol. Adv.*, **31**, 38–49.

Brezovsky,J. *et al.* (2016) Engineering a de novo transport tunnel. *ACS Catal.*, **6**, 7597–7610.

Chang,D.T.-H. *et al.* (2005) MEDock: a web server for efficient prediction of ligand binding sites based on a novel optimization algorithm. *Nucleic Acids Res.*, **33**, W233–W238.

Chovancova,E. *et al.* (2012) CAVER 3.0: a tool for the analysis of transport pathways in dynamic protein structures. *PLoS Comput. Biol.*, **8**, e1002708.

Clouthier,C.M. and Pelletier,J.N. (2012) Expanding the organic toolbox: a guide to integrating biocatalysis in synthesis. *Chem. Soc. Rev.*, **41**, 1585.

Cortes,J. *et al.* (2007) Molecular disassembly with Rrt-like algorithms. In: *Proceedings 2007 IEEE International Conference on Robotics and Automation, Rome, Italy*. IEEE, pp. 3301–3306.

Cui,Y.-L. *et al*. (2015) Molecular basis of the recognition of arachidonic acid by cytochrome P450 2E1 along major access tunnel. *Biopolymers*, **103**, 53–66.

Damborsky,J. and Brezovsky,J. (2009) Computational tools for designing and engineering biocatalysts. *Curr. Opin. Chem. Biol*., **13**, 26–34.

Daniel,L. *et al*. (2015) Mechanism-based discovery of novel substrates of haloalkane dehalogenases using *in silico* screening. *J. Chem. Inf. Model*., **55**, 54–62.

Devaurs,D. *et al*. (2013) MoMA-LigPath: a web server to simulate protein-ligand unbinding. *Nucleic Acids Res*., **41**, W297–W302.

Filipovic,J. *et al*. (2019) CaverDock: a novel method for the fast analysis of ligand transport. *IEEE/ACM Trans. Comput. Biol. Bioinform*., 1–1. doi: 10.1109/TCBB.2019.2907492.

Gora,A. *et al*. (2013) Gates of enzymes. *Chem. Rev*., **113**, 5871–5923.

Grubmüller,H. *et al*. (1996) Ligand binding: molecular mechanics calculation of the streptavidin-biotin rupture force. *Science*, **271**, 997–999.

Halgren,T.A. (1996) Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. *J. Comput. Chem*., **17**, 490–519.

Hanwell,M.D. *et al*. (2012) Avogadro: an advanced semantic chemical editor, visualization, and analysis platform. *J. Cheminform*., **4**, 17.

Hu,H. *et al*. (2016) Transient kinetics define a complete kinetic model for protein arginine methyltransferase 1. *J. Biol. Chem*., **291**, 26722–26738.

Irwin, J.J. and Shoichet, B.K. (2005) ZINC: a free database of commercially available compounds for virtual screening. *J. Chem. Inf. Model*., **45**, 177–182.

Kaushik,S. *et al*. (2018) Impact of the access tunnel engineering on catalysis is strictly ligand-specific. *Febs J*., **285**, 1456–1476.

Kingsley,L.J. and Lill,M.A. (2014) Including ligand-induced protein flexibility into protein tunnel prediction. *J. Comput. Chem*., **35**, 1748–1756.

Kmunícek,J. *et al*. (2005) Quantitative analysis of substrate specificity of haloalkane dehalogenase LinB from *Sphingomonas paucimobilis* UT26. *Biochemistry*, **44**, 3390–3401.

Koeller,K.M. and Wong,C.-H. (2001) Enzymes for chemical synthesis. *Nature*, **409**, 232–240.

Koudelakova,T. *et al*. (2011) Substrate specificity of haloalkane dehalogenases. *Biochem. J*., **435**, 345–354.

Koudelakova,T. *et al*. (2013) Engineering enzyme stability and resistance to an organic cosolvent by modification of residues in the access tunnel. *Angew. Chem. Int. Ed*., **52**, 1959–1963.

Lee,P.-H. *et al*. (2009) SLITHER: a web server for generating contiguous conformations of substrate molecules entering into deep active sites of proteins or migrating through channels in membrane transporters. *Nucleic Acids Res*., **37**, W559–W564.

Liskova,V. *et al*. (2017) Different structural origins of the enantioselectivity of haloalkane dehalogenases toward linear β-haloalkanes: open-solvated versus occluded-desolvated active sites. *Angew. Chem. Int. Ed*., **56**, 4719–4723.

Lüdemann,S.K. *et al*. (2000) How do substrates enter and products exit the buried active site of cytochrome P450cam? 1. Random expulsion molecular dynamics investigation of ligand access channels and mechanisms. *J. Mol. Biol*., **303**, 797–811.

Milani,M. *et al*. (2005) Structural bases for heme binding and diatomic ligand recognition in truncated hemoglobins. *J. Inorg. Biochem*., **99**, 97–109.

Mittermaier,A. and Meneses,E. (2013) Analyzing protein–ligand interactions by dynamic NMR spectroscopy. In: John,W. (ed.) *Methods in Molecular Biology*, Springer, Clifton, NJ, pp. 243–266.

Morris,G.M. *et al*. (2009) AutoDock4 and AutoDockTools4: automated docking with selective receptor flexibility. *J. Comput. Chem*., **30**, 2785–2791.

Peräkylä,M. (2009) Ligand unbinding pathways from the vitamin D receptor studied by molecular dynamics simulations. *Eur. Biophys. J*., **38**, 185–198.

Prokop,Z. *et al*. (2003) Catalytic mechanism of the maloalkane dehalogenase LinB from *Sphingomonas paucimobilis* UT26. *J. Biol. Chem*., **278**, 45094–45100.

Rydzewski,J. and Nowak,W. (2017) Ligand diffusion in proteins via enhanced sampling in molecular dynamics. *Phys. Life Rev*., **22–23**, 58–74.

de Sanctis,D. *et al*. (2004) Crystal structure of cytoglobin: the fourth globin type discovered in man displays heme hexa-coordination. *J. Mol. Biol*., **336**, 917–927.

Schmidt,M. *et al*. (2005) Ligand migration pathway and protein dynamics in myoglobin: a time-resolved crystallographic study on L29W MbCO. *Proc. Natl. Acad. Sci. USA*, **102**, 11704–11709.

Schomburg,I. *et al*. (2004) BRENDA, the enzyme database: updates and major new developments. *Nucleic Acids Res*., **32**, 431D–4433.

Schotte,F. *et al*. (2003) Watching a protein as it functions with 150-ps time-resolved X-ray crystallography. *Science*, **300**, 1944–1947.

Soetaert,W. and Vandamme,E. (2006) The impact of industrial biotechnology. *Biotechnol. J*., **1**, 756–769.

Tilton,R.F. *et al*. (1984) Cavities in proteins: structure of a metmyoglobin-xenon complex solved to 1.9 A. *Biochemistry*, **23**, 2849–2857.

Trott,O. and Olson,A.J. (2010) AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem*., **31**, 455–461.

The UniProt Consortium. (2017) UniProt: the universal protein knowledgebase. *Nucleic Acids Res*, **45**, D158–D169.

Wang,Y. *et al*. (2005) What makes an aquaporin a glycerol channel? A comparative study of AqpZ and GlpF. *Structure*, **13**, 1107–1118.

Caver Web 1.0: Identification of Tunnels and Channels in Proteins and Analysis of Ligand Transport.

Nucleic Acids Research, 2019 **1**
doi: 10.1093/nar/gkz378

Caver Web 1.0: identification of tunnels and channels in proteins and analysis of ligand transport

Jan Stourac[1,2], Ondrej Vavra[1,2], Piia Kokkonen[1], Jiri Filipovic[3], Gaspar Pinto[1,2], Jan Brezovsky[1], Jiri Damborsky[1,2] and David Bednar[1,2,*]

[1]Loschmidt Laboratories, Department of Experimental Biology and RECETOX, Faculty of Science, Masaryk University, Brno, Czech Republic, [2]International Centre for Clinical Research, St. Anne's University Hospital Brno, Brno, Czech Republic and [3]Institute of Computer Science, Masaryk University, Brno, Czech Republic

Received February 21, 2019; Revised April 24, 2019; Editorial Decision April 29, 2019; Accepted May 05, 2019

ABSTRACT

Caver Web 1.0 is a web server for comprehensive analysis of protein tunnels and channels, and study of the ligands' transport through these transport pathways. Caver Web is the first interactive tool allowing both the analyses within a single graphical user interface. The server is built on top of the abundantly used tunnel detection tool Caver 3.02 and CaverDock 1.0 enabling the study of the ligand transport. The program is easy-to-use as the only required inputs are a protein structure for a tunnel identification and a list of ligands for the transport analysis. The automated guidance procedures assist the users to set up the calculation in a way to obtain biologically relevant results. The identified tunnels, their properties, energy profiles and trajectories for ligands' passages can be calculated and visualized. The tool is very fast (2–20 min per job) and is applicable even for virtual screening purposes. Its simple setup and comprehensive graphical user interface make the tool accessible for a broad scientific community. The server is freely available at https://loschmidt.chemi.muni.cz/caverweb.

INTRODUCTION

Proteins are biomolecules responsible for a vast variety of functions in all living organisms. They serve as a building material of cells and participate in regulation, signalling, transport, and enzymatic catalysis of small molecules. From the structural point of view, proteins consist of one or more peptide chains forming highly complex 3D structures containing many internal clefts, grooves, protrusions and voids (1). Even though such empty spaces are disadvantageous from the stability point of view, in many proteins they form functionally important local substructures, such as active sites, binding sites, allosteric sites, tunnels and channels (2, 3). Anatomies and properties of these substructures significantly influence protein functions (3). In this study, we are interested in transport pathways for small ligands represented by protein tunnels and channels. The channels are typically characterized by two openings connecting different cellular environments and play a key role in the transport of various ions and small molecules through biomembranes. The tunnels are mainly present in globular proteins with catalytic function (enzymes) and serve as the access pathways for substrates, products, co-factors, water molecules and/or inhibitors from a bulk solvent to buried active sites. They can also connect two distinct active sites within a single protein. It has been experimentally demonstrated that the tunnels and their properties can define many important protein characteristics like substrate specificity, enantioselectivity, stability and activity (4–8). Therefore, the understanding of the transport pathways, their properties and impact on ligands' passage is important for deciphering the protein function as well as for practical applications in the fields of protein engineering and drug design.

The study of access pathways and ligand transport processes using experimental techniques is far from trivial. A quantitative description of these processes is usually obtained indirectly using transient kinetic measurements. The few available direct methods such as time-resolved crystallography and crystallography under xenon pressure are time-demanding and provide only specific information (9,10). Therefore, the function of tunnels and channels are often studied *in silico*. The tunnel and channel detection is already well a developed field (11–14). Most of the recent tools, for example Caver 3.02 (15), MolAxis 1.0 (16), Mole 2.0 (17), are based on the pathway detection in the Voronoi diagram representation of a protein structure and offer high-quality results in short calculation time.

In silico analyses of ligand transport are challenging and the majority of methods are based on some implementation of molecular dynamics simulations (18–21). These implementations employ various enhanced sampling approaches like Random Accelerated Molecular Dynam-

*To whom correspondence should be addressed. Tel: +420 5 4949 6302; Fax: +420 5 4949 6302; Email: 222755@mail.muni.cz

ics (22), Steered Molecular Dynamics (23–25), Umbrella Sampling (26), Adaptive Sampling (27) or Metadynamics (26,28) and provide highly robust and accurate results. However, they are very time demanding, which prevents their usage in comparative studies or screening campaigns. Moreover, they usually require advanced knowledge of the modelling technique and a good understanding of the studied system. As an alternative, less accurate, but dramatically faster methods were developed. CaverDock 1.0 (29) and SLITHER 1.0 (30) are based on the iterative molecular docking along the tunnel, while MoMA-LigPath 1.0 (31,32) uses a robotic Manhattan-like RRT algorithm.

Here we present Caver Web 1.0, a novel web server for detection and comprehensive analysis of tunnels and channels in the protein structures. The server relies on the calculation of well-established and widely used tunnel detection software Caver 3.02. Moreover, Caver Web also integrates an explicit analysis of ligand transport through tunnels, which extends its use towards comparative studies and virtual screenings. The analysis of ligand transport is carried out by CaverDock 1.0, which provides a good trade-off between computation time and accuracy, while maintaining robustness of the workflow. A great care has been devoted to making the graphical user interface of Caver Web intuitive. The overall workflow is facilitated by robust default values of parameters and several automatic guiding mechanisms, which assist the users to correctly set up the calculation. Important results can be analysed and viewed directly in the visualization window. Three detailed tutorials cover typical use-cases, illustrating applicability of the tool for users with no prior knowledge of bioinformatics.

WORKFLOW

The basic workflow of the Caver Web tool is depicted in Figure 1. The first step of the calculation is the selection of a protein structure and its pre-treatment. The second step is a selection of a starting point for tunnel detection. Protein tunnels are identified and analysed in the third step, and optionally used to study the transport of selected ligand(s) in the fourth step.

Structure selection and pre-treatment

The only required input is a protein tertiary structure. It can be specified either by the Protein Data Bank (33) accession code or uploaded as a file in the PDB or the CIF format. Uploaded structures are automatically converted to PDB using RCSB MAXIT tool (https://sw-tools.rcsb.org/apps/MAXIT/index.html), since Caver does not natively support CIF format. The structures are usually deposited in the form of asymmetric units, which may not reflect their naturally occurring quaternary forms (biological units) and an analysis carried on this structure may lead to wrong results and even detection of non-existing tunnels. To overcome this problem, MakeMultimer (http://watcut.uwaterloo.ca/tools/makemultimer/) is automatically executed for uploaded structures to detect their biological units. Their list and description are provided to users who can select the most appropriate biological unit or dismiss them and continue with the original structure.

Starting point selection

The most critical step in tunnel detection is the selection of a proper starting point. The position of this point constraints the Caver calculation and defines a common starting point for all detected tunnels. A wrongly positioned point can significantly affect the relevance of detected tunnels and even lead to irrelevant tunnels. To facilitate this selection, we designed several automated protocols that provide reliable starting points suitable for the most common scenarios. In enzymes, users are often interested in access pathways for ligands leading to active or binding sites. Thus, the best starting point for this analysis is usually placed inside the pocket containing the essential residues (catalytic pocket). Since there are many tools for pocket detection and several databases of essential residues, we implemented a fully automatic 'Catalytic pocket' mode, which combines pockets detection with the analysis of essential residues. Pockets are detected using Fpocket 2 (34), based on the search of alpha spheres in a Voronoi tessellation representation of protein structures and subsequent clustering of the spheres to larger elements. The advantage of this tool is that it provides a druggability score, which represents a likelihood that the drug-like molecules can bind to the pocket. The essential residues are obtained from the Mechanism and Catalytic Site Atlas (35) and SwissProt (36) databases. The entries in Mechanism and Catalytic Site Atlas are mapped using the PDB accession codes. The manually curated SwissProt database is searched using BLAST with the requirement of 30% sequence identity and sequence length between 90 and 110%. After essential residues are identified, the pockets are matched with these residues and the pockets containing at least one catalytic residue are marked as catalytic. If essential residues are missing, Caver Web offers two alternative helper modes. The first one lists all detected pockets and sorts them by the estimated druggability score. The second one places the starting point to the centre of mass of any ligand present in the structure. However, this mode requires that the protein was co-crystallized or soaked with ligands, which occupy the functional site of the protein. This mode should be used with a great care. Finally, Caver Web offers the possibility to calculate the position of the starting point based on the residues selected by the user in the protein sequence, which can be further adjusted by the manual optimization of coordinates.

Tunnel detection and analysis

Tunnel detection is carried out by Caver 3.02 (15) which searches for the paths with the given minimal radius and the lowest cost in the Voronoi tessellation representation of protein structures using Djikstra's algorithm and calculates their geometries, statistical properties and list of residues lining the tunnel and forming the bottleneck. Users can modify several important configuration parameters affecting the properties of the detected tunnels: (i) 'residues considered for tunnel calculation' are the parts of the structure which Caver will consider for the analysis to allow exclusion of the ligands, ions and water molecules; (ii) 'minimum probe radius' defines the minimal size of a spherical probe which must fit into the tunnel to be detected; (iii) 'shell depth' specifies the maximal depth of a surface region, i.e.

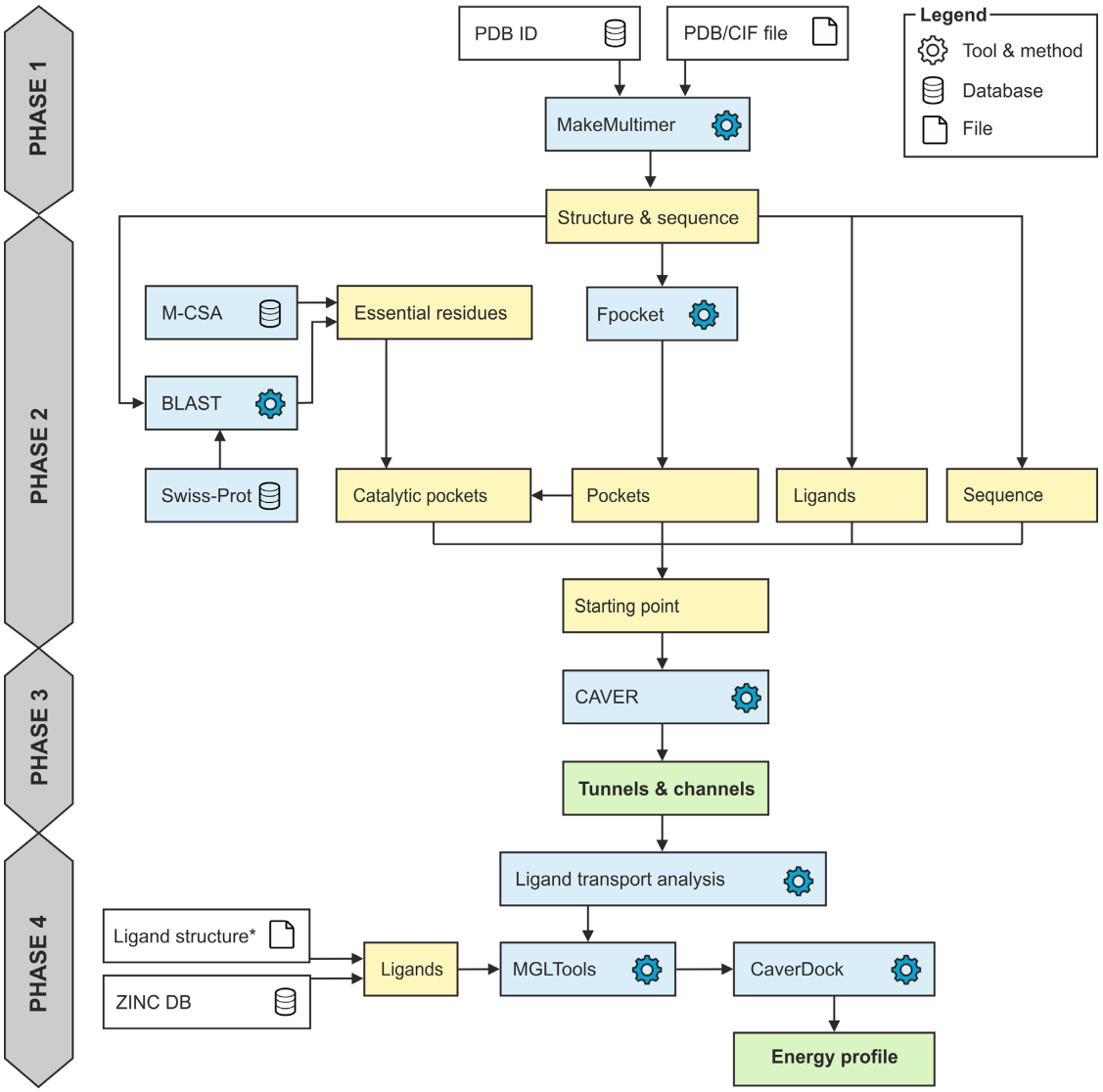

Figure 1. Workflow diagram of Caver Web 1.0. The process consists of four phases: (1) structure pre-treatment, (2) detection of the starting point, (3) identification of tunnels/channels and (4) analysis of ligand transport. *Ligand structures can be uploaded in all formats supported by Open Babel (http://openbabel.org/docs/current/FileFormats/Overview.html).

shallow vertices, preventing unnecessary tunnel branching; (iv) 'shell radius' specifies the radius of the probe used to define which parts of the Voronoi diagram represent the bulk solvent; (v) 'clustering threshold' defines the similarity level at which the tunnels will be considered the same and clustered together; (vi) 'maximal distance' which limits how far the starting Voronoi vertex can be from the starting point position selected by the user and (vii) 'desired radius' which specifies how far the starting point vertex must be from the atoms of the protein structure.

Ligand transport analysis

The last [optional] step of the workflow is the analysis of ligands transport through the detected tunnels using the CaverDock software. Initially, one or more small molecules must be provided by the user. Secondly, one or more identified tunnels are selected as the path for the ligand transport and a calculation is initiated. Caver Web adds Gastaiger charges and AutoDock Vina (37,38) compatible atom types to every atom using prepare_ligand4.py and prepare_receptor4.py scripts from the MGLTools (37) package. Then the Discretizer (29) is used to cut the tunnel to discrete slices with specified distances. Next, the CaverDock is executed to perform an iterative docking of the ligand to every slice of the tunnel using a spatially restrained AutoDock Vina docking algorithm.

Users can modify two most important parameters: (i) 'discretization delta' defines the distance between centres of two slices of the tunnel and (ii) 'calculation mode' of CaverDock defines which ligand restraints will be enforced. The first mode is called lower-bound and it enforces only the spatial restraint. This mode is very fast, however, it can miss some of the bottlenecks due to the possibility of ligand flipping, resulting in non-continuous movement. The second mode is called upper-bound and employs also the maxi-

mal ligand rotation restriction coupled with backtracking to guarantee continuous movement. Even though the continuous movements are more realistic, the analysis is computationally much more intensive and due to the limited capability of the backtracking it can overestimate energies or even completely fail to find any possible path. Therefore, the lower-bound trajectory is set as a default and users are strongly advised to use energetic profiles calculated in this mode. CaverDock supports flexible sidechains of selected residues. However, it has been shown that the energies of barriers are often artificially flattened (29), making the results difficult to interpret. For this reason, we suppressed the flexibility support in Caver Web until this issue is better resolved in future versions of CaverDock.

DESCRIPTION OF THE WEB SERVER

Input

The only mandatory input is the tertiary protein structure, which can be either specified by the accession code to the Protein Data Bank database or uploaded as a file in the PDB or the CIF format (Figure 2A). Once the structure is loaded, the MakeMultimer tool is automatically executed to detect the biological units. More details about each unit and their image preview can be shown by clicking the 'book' icon available on each row. The generated PDB file containing the biological unit can also be downloaded using the 'download' icon.

The next step is the selection of the starting point for the tunnel detection (Figure 2B). The page integrates the JSmol (39) molecular viewer which provides a visualization support to all modes and allows an immediate and interactive check of the current starting point position (represented as a red ball). Currently, we support four modes, available via separated tabs. The 'Catalytic pocket' mode is suitable for enzymes and combines detected pockets with essential residues obtained from the Mechanism and Catalytic Site Atlas and the SwissProt databases. For each catalytic pocket, a list of assigned essential residues, pocket relevance score, volume and the estimated druggability score are available. Once a particular pocket is selected, all surrounding residues are visualized as sticks and the pocket shape is represented as an isosurface. The position of the starting point is calculated as the average centre of mass of all residues of the selected pocket. The second mode 'Pocket' allows users to start from any detected pocket making it useful in the case when there are no essential residues available in the databases. By default, only top ten pockets are shown and ordered by their relevance. The rest is available on demand. Each pocket is described by its relevance, volume, and estimated druggability score. Furthermore, users can view the residues surrounding the pocket in the protein sequence. The starting point position is then calculated in the same way as for the 'Catalytic pocket' mode. The third mode is 'Ligand' and provides the possibility to place the starting point to the centre of the mass of any bound ligand. Each ligand is described using the formula, the name and the residue number. All ligands are visualized in sticks and distinguished using the different colours. The 'Sequence' mode allows users to select residues manually either from the sequence or directly from the visualized structure. Each selected residue is automatically visualized as sticks. The starting point position is calculated as the average centre of mass of selected residues. The 'Manual tuning' can be activated in all four cases of the starting point selection to adjust the x, y and z coordinates. Once the starting point is selected, users can adjust the parameters of Caver calculation. The parameters were described in the Workflow section.

Output of tunnel analysis

Users can specify a preferred job title for an easier orientation among submitted jobs. Notifications about the status of calculations can be sent to a provided email address. All jobs are stored and are accessible at any time. Once a job is submitted, tunnels are calculated using Caver tool and an analysis page is displayed. This page is divided into four major sections described below.

Job information. This section provides basic information about the job such as the identifier and the title. It also allows the user to directly download several files: (i) 'PyMOL session' downloads a pre-generated session file for the popular visualization software PyMOL. It contains the uploaded protein structure and all the detected tunnels offering the user to perform a detailed visual analysis or generate publication-quality images. (ii) 'Results zip' downloads an archive containing raw data generated by Caver during the calculation. The data can be used for advanced analyses or they can be directly imported to Caver Analyst (40). (iii) 'Caver configuration' opens a pop-up window with a complete configuration file used for the calculation. (iv) 'Caver log' opens a pop-up window with a raw textual output of Caver and provides details about the calculation process.

Tunnels info. The 'Tunnels info' section lists all identified tunnels and their selected properties (Figure 2C): (i) 'bottleneck radius' provides the maximal probe size which can fit in the narrowest part of the tunnel; (ii) 'length' quantifies the length of the tunnel from the starting point to the protein surface; (iii) 'curvature' describes the shape of the tunnel as the ratio between the length of the tunnel and the shortest possible distance between the starting point and the tunnel ending point; and (iv) 'throughput' reflects the probability that the pathway is used as a route for transport of the substances using the formula e^{-cost}, where e is Euler's number and the cost is a function defined as:

$$\int_0^L r(l)^{-2}\, dl$$

where L is a length of path, $r(l)$ is a function defining the radius of the largest ball which does not collide with the atoms of the structure and is centred at the point on the pathway axis in the distance l from the starting vertex (15). Every tunnel can be visualized by ticking the relevant checkbox and zoomed via the magnifying glass icon. Using the 'book' and the 'chart' icons, the 'Tunnel details' and the 'Tunnel profile' pop-up windows can be opened.

Tunnel details. The 'Tunnel details' pop-up window (Figure 2D) is organized into four tabs: (i) 'Overview' contains the important properties of the tunnel and a static picture

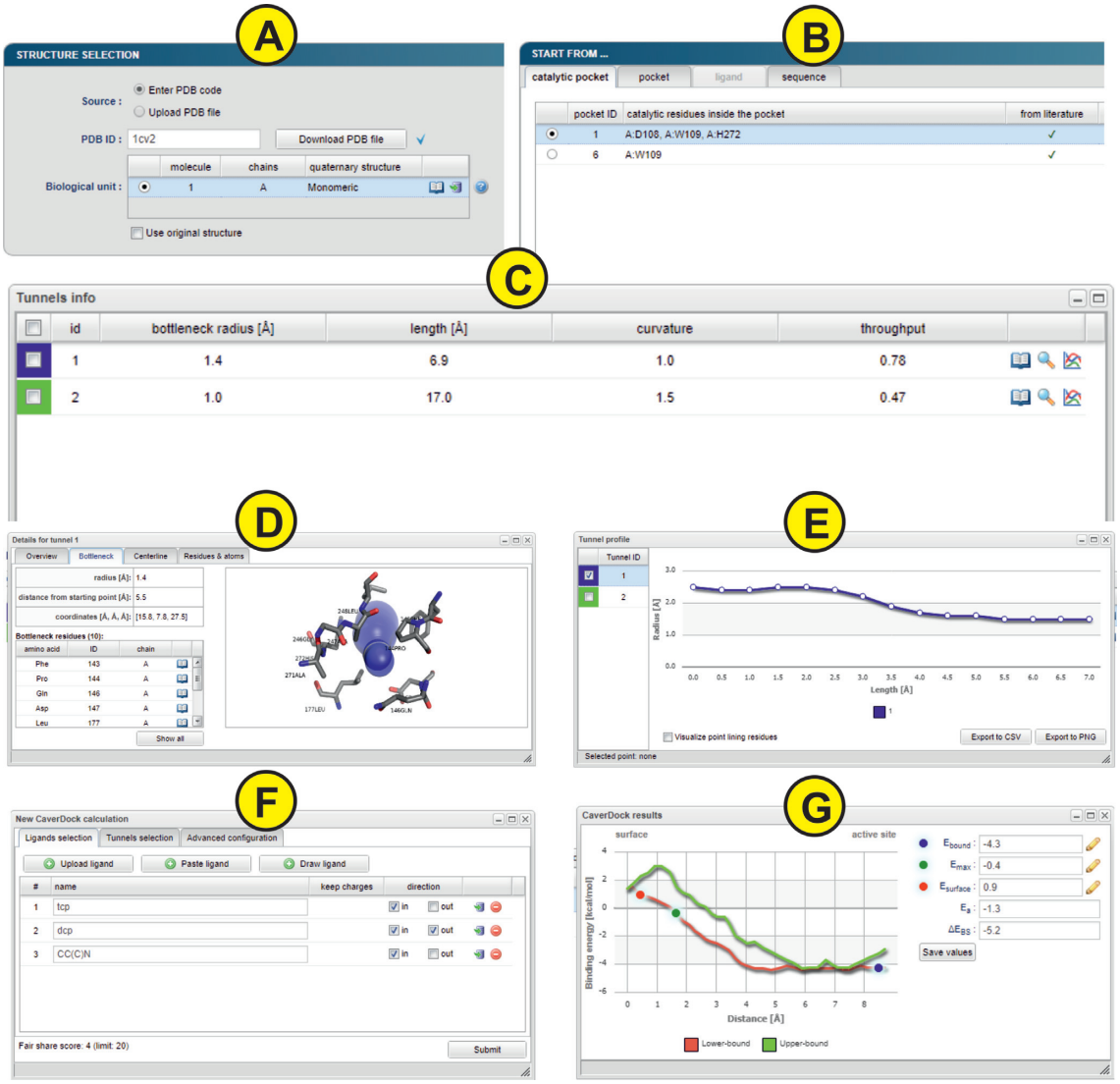

Figure 2. The graphical user interface of the Caver Web 1.0. The figure presents inputs and outputs obtained for the enzyme haloalkane dehalogenase LinB (PDB ID: 1CV2). (**A**) The 'Select structure' panel shows detected biological units for the provided protein structure. (**B**) The 'Starting point' panel for tunnel detection can be selected using four different methods. (**C**) The 'Tunnel info' panel provides an overview of the detected tunnels. (**D**) The 'Tunnel details' pop-up window presents detailed information about the selected tunnel. (**E**) The 'Tunnel profile' pop-up window shows the radius profile of the selected tunnels. (**F**) The 'New CaverDock calculation' pop-up window allows users to perform ligand transport analyses. (**G**) The 'CaverDock results' pop-up window displays calculated energy profiles for the selected ligand.

containing the protein as a cartoon and the tunnel visualized by spheres; (ii) 'Bottleneck' contains details about the narrowest part of the tunnel (bottleneck) including a list of surrounding residues and a static picture of the bottleneck with the tunnel visualized as spheres and surrounding residues as sticks; (iii) 'Centreline' lists all centres of the spheres along the tunnel centreline with their distance from the starting vertex on the Voronoi diagram, radius, coordinates of the centre and the Euclidean distance from the starting point; (iv) 'Residues & atoms' contains the list of all residues surrounding the tunnel.

Tunnel profile. The 'Tunnel profile' pop-up window (Figure 2E) allows a comparative analysis of tunnel profiles, i.e., the tunnel radius over the distance along the tunnel centreline. Users can select one or more tunnels from the table on the left and the graphs are automatically generated. Moreover, every data point is interactive and allows a selection of the proper tunnel sphere in the visualization. The displayed graphs can be downloaded either as CSV files or PNG images.

Protein visualization. The protein and all the detected tunnels can be interactively visualized directly in the web browser using the JSmol applet. Users can choose to visualize the protein structures using several commonly used visualization styles, display a starting point and a starting pocket, show detected tunnels as balls or line, and visualize their neighbouring residues.

Input of analysis of ligand transport

The last section of the output page from the tunnel calculation is devoted to an [optional] analysis of ligands transport through the tunnels.

Ligand transport analyses. The 'Ligand transport analyses' panel lists all CaverDock calculations with the basic information about the selected ligand, tunnel and the direction of the passage: (i) in – from the bulk solvent to the active site and (ii) out – from the active site to the bulk solvent. The status of each job is indicated as an icon - a green tick for successfully finished jobs, the 'zzz' icon for jobs waiting in a queue, an animated circle for currently running jobs and a red cross for failed jobs. More details about the job can be displayed by clicking on the 'book' icon. The log file containing all outputs generated during the calculation can be viewed using the 'text file' icon. Raw data can be directly downloaded using the 'download' icon. The 'Export data' button generates an Excel workbook with a summary sheet as well as a separate sheet for each job containing calculated energies (named by their identifier). A PDF report containing information about tunnels, jobs and energy plots can be generated by clicking on the 'Generate report' button.

Start new calculation. The 'Start new calculation' pop-up window (Figure 2F) is divided into three tabs: 'Ligands selection', 'Tunnels selection' and 'Advanced configuration'. In the first tab, users have three ways of providing the only mandatory input: (i) 'Upload ligand' allows the user to upload the ligand in any format supported by the Open Babel (41); (ii) 'Paste ligand' supports the input either in SMILES format or as an accession code to ZINC15 database (42) and (iii) 'Draw ligand' provides the possibility to draw ligand's structure manually using the interactive molecular editor JSME (43). Users can specify a preferred name for each ligand and a desired direction 'in' or 'out' of the active site. Molecules uploaded in mol2 format can also keep their original charges. The second tab contains the list of all tunnels and allows the user to make their selection for the analysis. The last tab allows a modification of two parameters for the CaverDock calculation: 'Discretization delta' and 'Calculation mode' which were described in the Workflow section. Users can also select the ligands that should be kept in the structure during the analysis. The residue names considered during the tunnel detection are automatically selected by default. Since users can upload multiple ligands and select multiple tunnels, the submission can easily lead to a combinatorial explosion. To ensure fairness among users and prevent overloading of the computational resources, the number of concurrently running calculations is limited using a fair share score: $F = F_C + (L_{IN} + L_{OUT}) * T * M$, where F_C is the fair share of currently running jobs, L_{IN} and L_{OUT} is the number of ligands passing in and out, respectively, T is the number of tunnels and M is the calculation mode coefficient (1 for lower-bound calculation, 1.5 for upper-bound calculation).

Output of ligand transport analysis

Energy profile. The 'Energy profile' pop-up window (Figure 2G) shows the graph of the calculated binding energies for each disc. Furthermore, the window also enables an automatic calculation of the activating energy and the energy difference between ligand bound on the surface and in the active site. The users have to interactively select three points from the graph: (i) E_B – the energy minimum of the ligand bound in the active site; (ii) E_{MAX} – the maximum energy of the transition and (iii) E_S – the energy minimum of ligand bound in the tunnel mouth. The 'Save values' button stores the values in the report file.

Generate report. The 'Generate report' pop-up window is a configuration dialog allowing users to adapt the content and the format of the report. It is divided into two tabs. The first one contains the list of all successfully finished jobs allowing users to select which jobs should be included in the report. The second tab focused on energy profiles enables user selection of the scaling mode of all graph axes (trajectory, energy, and tunnel radius): (i) 'Automatic' scales the axis based on the minimal and maximal values of each job separately; (ii) 'Automatic normalization' scales the axis based on the minimal and maximal values for all selected jobs and (iii) 'Manual limits' scales the axis to the manually entered values.

Use cases

The Caver Web tool can be used to address various biochemical problems. Three tutorials presented here and on the web portal provide an overview how Caver Web can be used: (i) to compare tunnels of different enzymes, (ii) to compare the passage of ligands via different tunnels of an enzyme and (iii) to screen a library of ligands for their passage through tunnels.

Case 1. Comparing the access tunnels of haloalkane dehalogenases. A comparison of protein tunnels can provide new insights into the structural elements coding for functional differences (2,11,44). Here, we studied the tunnels of five haloalkane dehalogenases (LinB, DmmA, DbjA, DhaA and DhlA), which catalyze the cleavage of carbon–halogen bonds in various halogenated hydrocarbons. These enzymes are closely related and their catalytic residues are conserved, yet their substrate preferences vary significantly (45,46). With the Caver Web tool we can show that the enzymes with more constricted tunnels (bottleneck < 1.5 Å) tend to be most effective with small substrates, e.g., DhlA with 1,2-dichloroethane and LinB with 1,2-dibromoethane. DmmA with the widest tunnels (bottleneck 2.5 Å) prefers the larger substrate 4-bromobutanenitrile. Conformational changes will be needed for binding of larger molecules to haloalkane dehalogenases via narrow tunnels (47).

Case 2. Studying paracetamol binding to the human cytochrome P450 3A4. Human cytochrome P450 enzymes (CYPs) metabolize a wide range of different substrates. The enzymes show a broad substrate specificity and possess multiple tunnels leading from the protein surface to the catalytic site. CYP3A4 is the main drug metabolizing enzyme in the liver, participating in the metabolism of ∼30% of available drugs (48,49). One of its substrates, paracetamol, is a common analgesic and antipyretic drug. Caver Web calculations revealed that the most preferred route for paracetamol

Table 1. Comparison of Caver Web with available servers for detection of tunnels and channels in proteins and ligand transport analysis. Caver Web is currently the only tool which provides a one-stop shop for tunnel/channel identification and analysis of transport processes. Comprehensive comparison of Caver and CaverDock with other tools can be found in their primary publications (15,29).

		Tunnels and channels analysis		Ligand transport analysis			
Software	Input	Supported	Starting point selection	Supported	Ligand source	Output	Ref.
Caver Web	PDB ID[b], PDB/CIF file[b]	Yes	Catalytic pocket, pocket, ligands, residues, coordinates	Yes	ZINC15, user file, drawing	Tunnels/channels, ligand trajectory, energy profile	this study
MolAxis	PDB ID, PDB file	Yes	Largest void, coordinates	No	-[d]	Tunnels/channels	(16)
MoleOnline	PDB ID[c], CIF/PDB file[c]	Yes	Catalytic residues, residues, coordinates, pocket, pattern	No	-[d]	Tunnels/channels	(51)
BetaCavityWeb	PDB ID, PDB file	Yes	*Not required*	No	-[d]	Tunnels/channels	(52)
PoreWalker	PDB file	Yes	*Not required*	No	-[d]	Channels	(12)
ChExVis	PDB ID, PDB file	Yes	Catalytic residues, HETATM records, residues	No	-[d]	Tunnels/channels	(53)
MoMA-LigPath[a]	PDB file	No	-[d]	Yes	Part of PDB file	Ligand trajectory	(32)

[a]Web server SLITHER for ligand transport analysis was not accessible in the time of writing.
[b]Biological units detection by MakeMultimer.
[c]Biological units fetched from the PDBe database (54).
[d]Not applicable.

binding to CYP3A4 is via the tunnel #2. Paracetamol can also bind through the tunnel #3, while its binding through the tunnels #1 and #4 requires conformational changes.

Case 3. Virtual screening of leukotriene A4 hydrolase/aminopeptidase inhibitors. Virtual screening is a well-established technique for drug design and there are many web services available for this purpose (50). Caver Web enables docking of ligands along a tunnel. This procedure significantly enhances the sampling region as compared to the classical docking. Our target the leukotriene A4 hydrolase/aminopeptidase (EC 3.3.2.6), is a bifunctional zinc metalloenzyme that catalyses the formation of the chemotactic agent LTB4, a key lipid mediator in the immune response. We screened 21 ligands and the resulting binding energy profiles were used for ligand ranking. We found out that the inhibitors ibuprofen and flurbiprofen have the easiest passage through the main tunnel. An additional finding was that oxaprozin binds stronger inside the tunnel than in the active site, which might indicate an inhibition mechanism based on a tunnel blockage. Such information would not be available from a classical virtual screening study targeting only the active site.

CONCLUSIONS AND OUTLOOK

Caver Web 1.0 is a novel web server for structural and functional analysis of the tunnels and channels in protein structures. The tool complements tunnels and channels detection by an explicit analysis of ligand transport (Table 1). This unique functionality dramatically expands its use towards virtual screening in drug design applications. The server provides a simple and easy-to-use graphical user interface. Importantly, Caver Web integrates several automated helper procedures that guide the users through the workflow. They assist a correct setup of the calculation without a deep understanding of the setup and navigate the interpretation of the data obtained by the individual integrated tools. Caver Web improves the results of virtual screenings by analyzing the ability of potential inhibitors to reach their binding positions. The limitations of the web server relate to its simple interface. Some of the advanced analyses offered by the stand-alone versions of the software could be difficult to conduct via the web interface. Moreover, an analysis of extensive datasets, such as large libraries of ligands or protein assemblies from molecular dynamic trajectories, is also restricted due to the available computational resources.

New features will be implemented in the future versions of Caver Web. Firstly, we plan to optimize the position of the starting point within the pockets. The current algorithm places the point in the middle of the pocket, which in some cases leads to a shortening of the tunnel length. Therefore, we will develop a new algorithm, which will automatically push the starting point deeper into the pocket. Secondly, we will focus on protein dynamics, which can be crucial for efficient ligand transport through access tunnels in many biological systems. An incorporation of the side chains' flexibility or an analysis of molecular ensembles can provide important insights into the tunnel dynamics and their importance for transport processes. Thirdly, a possibility to introduce mutations to tunnel-lining or bottleneck residues and then to recalculate analyses will expand in protein engineering. Finally, the currently used visualization tool JSmol will be replaced by the Mol* tool, which is being developed by PDBe and RCSB PDB teams.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

FUNDING

Czech Ministry of Education [LM2015047, LM2015055, CZ.02.1.01/0.0/0.0/16_013/0001761, CZ.02.1.01/0.0/0.0/16_026/0008451]; Raft4Biotech and SinFonia from the European Union [720776, 814418]; The MetaCentrum and CERIT-SC are acknowledged for providing access to computing facilities [LM2010005, CZ.1.05/3.2.00/08.0144]. Funding for open access charge: Czech Ministry of Education.

Conflict of interest statement. None declared.

REFERENCES

1. Calland,P.-Y. (2003) On the structural complexity of a protein. *Protein Eng. Des. Sel.*, **16**, 79–86.
2. Kingsley,L.J. and Lill,M.A. (2015) Substrate tunnels in enzymes: Structure–function relationships and computational methodology. *Proteins Struct. Funct. Bioinf.*, **83**, 599–611.
3. Biedermannová,L., Prokop,Z., Gora,A., Chovancová,E., Kovács,M., Damborský,J. and Wade,R.C. (2012) A single mutation in a tunnel to the active site changes the mechanism and kinetics of product release in haloalkane dehalogenase LinB. *J. Biol. Chem.*, **287**, 29062–29074.
4. Richards,F.M. (1997) Protein stability: still an unsolved problem. *Cell. Mol. Life Sci.*, **53**, 790–802.
5. Kaushik,S., Prokop,Z., Damborsky,J. and Chaloupkova,R. (2017) Kinetics of binding of fluorescent ligands to enzymes with engineered access tunnels. *FEBS J.*, **284**, 134–148.
6. Kaushik,S., Marques,S.M., Khirsariya,P., Paruch,K., Libichova,L., Brezovsky,J., Prokop,Z., Chaloupkova,R. and Damborsky,J. (2018) Impact of the access tunnel engineering on catalysis is strictly ligand-specific. *FEBS J.*, **285**, 1456–1476.
7. Brezovsky,J., Babkova,P., Degtjarik,O., Fortova,A., Gora,A., Iermak,I., Rezacova,P., Dvorak,P., Smatanova,I.K., Prokop,Z. *et al.* (2016) Engineering a de Novo Transport Tunnel. *ACS Catal.*, **6**, 7597–7610.
8. Liskova,V., Bednar,D., Prudnikova,T., Rezacova,P., Koudelakova,T., Sebestova,E., Smatanova,I.K., Brezovsky,J., Chaloupkova,R. and Damborsky,J. (2015) Balancing the stability–activity trade-off by fine-tuning dehalogenase access tunnels. *ChemCatChem*, **7**, 648–659.
9. Schmidt,M., Nienhaus,K., Pahl,R., Krasselt,A., Anderson,S., Parak,F., Nienhaus,G.U. and Šrajer,V. (2005) Ligand migration pathway and protein dynamics in myoglobin: a time-resolved crystallographic study on L29W MbCO. *Proc. Natl Acad. Sci. U.S.A.*, **102**, 11704–11709.
10. Šrajer,V., Ren,Z., Teng,T.-Y., Schmidt,M., Ursby,T., Bourgeois,D., Pradervand,C., Schildkamp,W., Wulff,M. and Moffat,K. (2001) Protein conformational relaxation and ligand migration in myoglobin: a nanosecond to millisecond molecular movie from Time-Resolved laue X-ray diffraction. *Biochemistry*, **40**, 13802–13815.
11. Brezovsky,J., Chovancova,E., Gora,A., Pavelka,A., Biedermannova,L. and Damborsky,J. (2013) Software tools for identification, visualization and analysis of protein tunnels and channels. *Biotechnol. Adv.*, **31**, 38–49.
12. Pellegrini-Calace,M., Maiwald,T. and Thornton,J.M. (2009) PoreWalker: a novel tool for the identification and characterization of channels in transmembrane proteins from their Three-Dimensional structure. *PLoS Comput. Biol.*, **5**, e1000440.
13. Voss,N.R. and Gerstein,M. (2010) 3V: cavity, channel and cleft volume calculator and extractor. *Nucleic Acids Res.*, **38**, W555–W562.
14. Oliveira,S.H., Ferraz,F.A., Honorato,R.V., Xavier-Neto,J., Sobreira,T.J. and de Oliveira,P.S. (2014) KVFinder: steered identification of protein cavities as a PyMOL plugin. *BMC Bioinformatics*, **15**, 197.
15. Chovancova,E., Pavelka,A., Benes,P., Strnad,O., Brezovsky,J., Kozlikova,B., Gora,A., Sustr,V., Klvana,M., Medek,P. *et al.* (2012) CAVER 3.0: a tool for the analysis of transport pathways in dynamic protein structures. *PLoS Comput. Biol.*, **8**, e1002708.
16. Yaffe,E., Fishelovitch,D., Wolfson,H.J., Halperin,D. and Nussinov,R. (2008) MolAxis: a server for identification of channels in macromolecules. *Nucleic Acids Res.*, **36**, W210–W215.
17. Sehnal,D., Svobodová Vařeková,R., Berka,K., Pravda,L., Navrátilová,V., Banáš,P., Ionescu,C.-M., Otyepka,M. and Koča,J. (2013) MOLE 2.0: advanced approach for analysis of biomacromolecular channels. *J. Cheminformatics*, **5**, 39.
18. Long,D., Mu,Y. and Yang,D. (2009) Molecular dynamics simulation of ligand dissociation from liver fatty acid binding protein. *PLoS One*, **4**, e6081.
19. Gu,Y., Shrivastava,I.H., Amara,S.G. and Bahar,I. (2009) Molecular simulations elucidate the substrate translocation pathway in a glutamate transporter. *Proc. Natl Acad. Sci. U.S.A.*, **106**, 2589–2594.
20. Kaus,J.W. and McCammon,J.A. (2015) Enhanced ligand sampling for relative protein–ligand binding free energy calculations. *J. Phys. Chem. B*, **119**, 6190–6197.
21. Doerr,S. and De Fabritiis,G. (2014) On-the-Fly learning and sampling of ligand binding by high-throughput molecular simulations. *J. Chem. Theory Comput.*, **10**, 2064–2069.
22. Kokh,D.B., Amaral,M., Bomke,J., Grädler,U., Musil,D., Buchstaller,H.-P., Dreyer,M.K., Frech,M., Lowinski,M., Vallee,F. *et al.* (2018) Estimation of Drug-Target residence times by τ-Random acceleration molecular dynamics simulations. *J. Chem. Theory Comput.*, **14**, 3859–3869.
23. Chen,L.Y. (2015) Hybrid steered molecular dynamics approach to computing absolute binding free energy of Ligand–Protein Complexes: A brute force approach that is fast and accurate. *J. Chem. Theory Comput.*, **11**, 1928–1938.
24. Do,P.-C., Lee,E.H. and Le,L. (2018) Steered molecular dynamics simulation in rational drug design. *J. Chem. Inf. Model.*, **58**, 1473–1482.
25. Skovstrup,S., David,L., Taboureau,O. and Jørgensen,F.S. (2012) A steered molecular dynamics study of binding and translocation processes in the GABA transporter. *PLoS One*, **7**, e39360.
26. Zhang,Y. and Voth,G.A. (2011) A combined metadynamics and umbrella sampling method for the calculation of ion permeation free energy profiles. *J. Chem. Theory Comput.*, **7**, 2277–2283.
27. Marques,S.M., Bednar,D. and Damborsky,J. (2019) Computational study of protein-ligand unbinding for enzyme engineering. *Front. Chem.*, **6**, 650.
28. Furini,S. and Domene,C. (2016) Computational studies of transport in ion channels using metadynamics. *Biochim. Biophys. Acta (BBA) - Biomembranes*, **1858**, 1733–1740.
29. Filipovič,J., Vávra,O., Plhák,J., Bednář,D., Marques,S.M., Brezovský,J., Matyska,L. and Damborský,J. (2019) CaverDock: a novel method for the fast analysis of ligand transport. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, doi:10.1109/TCBB.2019.2907492.
30. Lee,P.-H., Kuo,K.-L., Chu,P.-Y., Liu,E.M. and Lin,J.-H. (2009) SLITHER: a web server for generating contiguous conformations of substrate molecules entering into deep active sites of proteins or migrating through channels in membrane transporters. *Nucleic Acids Res.*, **37**, W559–W564.
31. Cortés,J., Siméon,T., Ruiz de Angulo,V., Guieysse,D., Remaud-Siméon,M. and Tran,V. (2005) A path planning approach for computing large-amplitude motions of flexible molecules. *Bioinformatics*, **21**, i116–i125.
32. Cortés,J., Le,D.T., Iehl,R. and Siméon,T. (2010) Simulating ligand-induced conformational changes in proteins using a mechanical disassembly method. *Phys. Chem. Chem. Phys.*, **12**, 8268–8276.
33. Burley,S.K., Berman,H.M., Bhikadiya,C., Bi,C., Chen,L., Costanzo,L.D., Christie,C., Duarte,J.M., Dutta,S., Feng,Z. *et al.* (2019) Protein Data Bank: the single global archive for 3D macromolecular structure data. *Nucleic Acids Res.*, **47**, D520–D528.
34. Le Guilloux,V., Schmidtke,P. and Tuffery,P. (2009) Fpocket: an open source platform for ligand pocket detection. *BMC Bioinformatics*, **10**, 168.
35. Ribeiro,A.J.M., Holliday,G.L., Furnham,N., Tyzack,J.D., Ferris,K. and Thornton,J.M. (2018) Mechanism and Catalytic Site Atlas (M-CSA): a database of enzyme reaction mechanisms and active sites. *Nucleic Acids Res.*, **46**, D618–D623.
36. The UniProt Consortium (2018) UniProt: the universal protein knowledgebase. *Nucleic Acids Res.*, **46**, 2699–2699.
37. Morris,G.M., Huey,R., Lindstrom,W., Sanner,M.F., Belew,R.K., Goodsell,D.S. and Olson,A.J. (2009) AutoDock4 and

AutoDockTools4: automated docking with selective receptor flexibility. *J. Comput. Chem.*, **30**, 2785–2791.
38. Trott,O. and Olson,A.J. (2010) AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *J. Comput. Chem.*, **31**, 455–461.
39. Hanson,R.M., Prilusky,J., Renjian,Z., Nakane,T. and Sussman,J.L. (2013) JSmol and the next-generation web-based representation of 3D molecular structure as applied to proteopedia. *Isr. J. Chem.*, **53**, 207–216.
40. Jurcik,A., Bednar,D., Byska,J., Marques,S.M., Furmanova,K., Daniel,L., Kokkonen,P., Brezovsky,J., Strnad,O., Stourac,J. *et al.* (2018) CAVER Analyst 2.0: analysis and visualization of channels and tunnels in protein structures and molecular dynamics trajectories. *Bioinformatics*, **34**, 3586–3588.
41. O'Boyle,N.M., Banck,M., James,C.A., Morley,C., Vandermeersch,T. and Hutchison,G.R. (2011) Open Babel: an open chemical toolbox. *J. Cheminformatics*, **3**, 33.
42. Sterling,T. and Irwin,J.J. (2015) ZINC 15 – ligand discovery for everyone. *J. Chem. Inf. Model.*, **55**, 2324–2337.
43. Bienfait,B. and Ertl,P. (2013) JSME: a free molecule editor in JavaScript. *J. Cheminformatics*, **5**, 24.
44. Marques,S.M., Daniel,L., Buryska,T., Prokop,Z., Brezovsky,J. and Damborsky,J. (2017) Enzyme tunnels and gates as relevant targets in drug design. *Med. Res. Rev.*, **37**, 1095–1139.
45. Koudelakova,T., Chovancova,E., Brezovsky,J., Monincova,M., Fortova,A., Jarkovsky,J. and Damborsky,J. (2011) Substrate specificity of haloalkane dehalogenases. *Biochem. J.*, **435**, 345–354.
46. Gehret,J.J., Gu,L., Geders,T.W., Brown,W.C., Gerwick,L., Gerwick,W.H., Sherman,D.H. and Smith,J.L. (2012) Structure and activity of DmmA, a marine haloalkane dehalogenase. *Protein Sci.*, **21**, 239–248.
47. Kokkonen,P., Bednar,D., Dockalova,V., Prokop,Z. and Damborsky,J. (2018) Conformational changes allow processing of bulky substrates by a haloalkane dehalogenase with a small and buried active site. *J. Biol. Chem.*, **293**, 11505–11512.
48. Haddad,A., Davis,M. and Lagman,R. (2007) The pharmacological importance of cytochrome CYP3A4 in the palliation of symptoms: review and recommendations for avoiding adverse drug interactions. *Support. Care Cancer*, **15**, 251–257.
49. Baylon,J.L., Lenov,I.L., Sligar,S.G. and Tajkhorshid,E. (2013) Characterizing the membrane-bound state of cytochrome P450 3A4: structure, depth of insertion, and orientation. *J. Am. Chem. Soc.*, **135**, 8542–8551.
50. Banegas-Luna,A.-J., Cerón-Carrasco,J.P. and Pérez-Sánchez,H. (2018) A review of ligand-based virtual screening web tools and screening algorithms in large molecular databases in the age of big data. *Future Med. Chem.*, **10**, 2641–2658.
51. Pravda,L., Sehnal,D., Toušek,D., Navrátilová,V., Bazgier,V., Berka,K., Svobodová Vareková,R., Koca,J. and Otyepka,M. (2018) MOLEonline: a web-based tool for analyzing channels, tunnels and pores. *Nucleic Acids Res.*, **46**, W368–W373.
52. Kim,J.K., Cho,Y., Lee,M., Laskowski,R.A., Ryu,S.E., Sugihara,K. and Kim,D.S. (2015) BetaCavityWeb: a webserver for molecular voids and channels. *Nucleic Acids Res.*, **43**, W413–W418.
53. Masood,T.B., Sandhya,S., Chandra,N. and Natarajan,V. (2015) CHEXVIS: a tool for molecular channel extraction and visualization. *BMC Bioinformatics*, **16**. 119–138.
54. Gutmanas,A., Alhroub,Y., Battle,G.M., Berrisford,J.M., Bochet,E., Conroy,M.J., Dana,J.M., Fernandez Montecelo,M.A., van Ginkel,G., Gore,S.P. *et al.* (2014) PDBe: Protein Data Bank in Europe.. *Nucleic Acids Res.*, **42**, D285–D291.

HotSpot Wizard 3.0: Web Server for Automated Design of Mutations and Smart Libraries Based on Sequence Input Information.

W356–W362 *Nucleic Acids Research, 2018, Vol. 46, Web Server issue* *Published online 23 May 2018*
doi: 10.1093/nar/gky417

HotSpot Wizard 3.0: web server for automated design of mutations and smart libraries based on sequence input information

Lenka Sumbalova[1,2], Jan Stourac[1,3], Tomas Martinek[2], David Bednar[1,3,*] and Jiri Damborsky[1,3,*]

[1]Loschmidt Laboratories, Department of Experimental Biology, Masaryk University, 62500 Brno, Czech Republic, [2]IT4Innovations Centre of Excellence, Faculty of Information Technology, Brno University of Technology, Bozetechova 2, 61266 Brno, Czech Republic and [3]International Centre for Clinical Research, St. Anne's University Hospital Brno, 65691 Brno, Czech Republic

Received February 04, 2018; Revised April 20, 2018; Editorial Decision May 02, 2018; Accepted May 07, 2018

ABSTRACT

HotSpot Wizard is a web server used for the automated identification of hotspots in semi-rational protein design to give improved protein stability, catalytic activity, substrate specificity and enantioselectivity. Since there are three orders of magnitude fewer protein structures than sequences in bioinformatic databases, the major limitation to the usability of previous versions was the requirement for the protein structure to be a compulsory input for the calculation. HotSpot Wizard 3.0 now accepts the protein sequence as input data. The protein structure for the query sequence is obtained either from eight repositories of homology models or is modeled using Modeller and I-Tasser. The quality of the models is then evaluated using three quality assessment tools—WHAT_CHECK, PROCHECK and MolProbity. During follow-up analyses, the system automatically warns the users whenever they attempt to redesign poorly predicted parts of their homology models. The second main limitation of HotSpot Wizard's predictions is that it identifies suitable positions for mutagenesis, but does not provide any reliable advice on particular substitutions. A new module for the estimation of thermodynamic stabilities using the Rosetta and FoldX suites has been introduced which prevents destabilizing mutations among pre-selected variants entering experimental testing. HotSpot Wizard is freely available at http://loschmidt.chemi.muni.cz/hotspotwizard.

INTRODUCTION

Proteins are macromolecules with many biological functions. Apart from their irreplaceable role in all living organisms, they are also widely used in many fields, including medicine (1), enzymology (2), synthetic biology (3) and material science (4). Naturally occurring proteins often do not meet the specifications for practical applications. Therefore, protein engineers modify sequences to obtain enhanced properties or completely new functions. Directed evolution, which has been an extremely successful protein engineering technology, does not require a molecular understanding of the impact of mutation on the protein structure (5). Modified proteins are generated in iterative rounds of mutation and screening or selection of the best hits that possess the required property (6). The obvious disadvantage to this method is that only a tiny fraction of all protein variants contain the desired property. Analysis of libraries containing millions of mutants is costly and time-consuming. Semi-rational protein engineering is an approach that implements *in silico* identification of important regions of the protein so that mutagenesis is better located, resulting in smaller high-quality libraries (7). The key step to semi-rational protein engineering is the selection of hotspot residues whose mutations will bring the largest improvement to the target protein properties (8).

HotSpot Wizard 2.0 (9) is an interactive web server used for the identification of hotspots in proteins by automated multi-step calculation and a comprehensive presentation of results. The tool makes protein design accessible to researchers with no prior knowledge of bioinformatics. After entering an input protein structure, 19 prediction tools and 3 databases are used for protein annotation. HotSpot Wizard then provides four different strategies for selecting hotspots: (i) functional hotspots corresponding to highly mutable residues located in the active site

*To whom correspondence should be addressed. Tel: +420 5 4949 3467; Fax: +420 5 4949 6302; Email: jiri@chemi.muni.cz
*Correspondence may also be addressed to David Bednar. Email: 222755@mail.muni.cz

pocket or access tunnels, (ii) stability hotspots corresponding to flexible residues, (iii) stability hotspots from back-to-consensus analysis and (iv) correlated hotspots corresponding to pairs of co-evolving residues. The users can design a smart library based on naturally accepted substitutions from phylogenetic analysis. HotSpot Wizard 2.0 (9) has been used for over 10 000 protein structures by more than 1000 unique users since its release. For example, HotSpot Wizard has been used for the design of smart libraries of oxyhaemoglobin protein (10), for analysis leading to thermostabilization of a xylanase (11) and for identification of hotspots in a mutagenesis study of the transcription factor DREB1A (12). Previous implementations of HotSpot Wizard had two major drawbacks: (i) a requirement for the tertiary structure as essential input information and (ii) identification of positions for mutagenesis without quantification of the effects of individual substitutions on protein stability. HotSpot Wizard 3.0 shows dramatically enhanced usability by overcoming both these key limitations.

There are about 135 000 protein structures available in the RCSB Protein Data Bank (13), but there are more than 98 000 000 known protein sequences (14). Usage of HotSpot Wizard 2.0 is limited to the proteins with an available 3D structure. A solution to this problem is the prediction of the protein structure from its sequence by comparative (homology) modeling or threading (15). Homology modeling is based on the fact that members of a protein family with similar sequences also have similar tertiary structures (16,17). In HotSpot Wizard 3.0, it is possible to enter a sequence for a protein and have its tertiary structure retrieved from the repositories of models or constructed *ad hoc*. As the quality of the protein structure is critical for further structure analyses carried out by HotSpot Wizard, a robust quality assessment of the protein structure is provided using three well-established tools. The current implementation of our web server predicts hot-spots for mutagenesis and designs smart libraries based on phylogeny, but does not provide any quantitative analysis of individual substitutions, which is important, for example, in studies analyzing structure–function relationships. Moreover, screening or selection for multiple mutations at several different positions can still be time-consuming and so pre-selection of the most appropriate mutations is desirable. To help our users rationally decrease the number of variants for experimental testing, protein stability prediction has been introduced to discard potentially destabilizing mutations.

MATERIALS AND METHODS

Searches of structural databases and model depositories

The overall workflow of HotSpot Wizard 3.0 is outlined in Figure 1. When a protein sequence is used as an input, HotSpot Wizard: (i) searches experimentally determined structures, (ii) searches computationally modeled structures and (iii) constructs a homology model. The first step in this workflow is searching the RCSB Protein Data Bank (13). In this phase, only protein structures with a 100% sequence identity match (or part of the sequence matching the input with 100% sequence identity) are provided as a starting structure for the analysis. If no such structure is found, the Protein Model Portal (18) is searched. The Protein Model Portal collates models of protein structures from eight different resources: Center for Structures of Membrane Proteins, CSMP (19), Joint Center for Structural Genomics, JCSG (20), Midwest Center for Structural Genomics, MCSG (21), Northeast Structural Genomics Consortium, NESG (22), New York SGX Research Center for Structural Genomics, NYSGXRC (23), Joint Center for Molecular Modeling, JCMM (24), ModBase (25) and SWISS-MODEL Repository (26). HotSpot Wizard queries the Protein Model Portal and then lists all available hits. After selection of one of these models, the structure is downloaded directly to Hotspot Wizard from the repository.

Homology modeling

Whenever a homology model is not found or the user is not satisfied with the quality of the models available in public depositories, HotSpot Wizard carries out the homology modeling during the phase 1 (Figure 1). There is a wide range of homology modeling tools available. Twelve tools were initially considered for our workflow: SWISS-MODEL (27), Rosetta (28), Robetta (29), PHYRE2 (30), Pcons (31), Modeller (32), I-Tasser (33), IntFold (34), IMP (35), HHPred (36), RaptorX (37) and Sparks-X (38). These tools were analyzed for their availability as well as performance using Continuous Automated Model Evaluation, CAMEO (18) and Critical Assessment of Protein Structure Prediction, CASP (39). These community-wide comparisons evaluate structure predictions with available experimental data. Based on results from CASP and CAMEO, six tools were selected for further consideration, installed locally and tested (Modeller, Sparks-X, RaptorX, Rosetta, I-Tasser and SWISS-MODEL). RaptorX is very accurate with good coverage (i.e. percentage of submitted models, which could be successfully modeled), but it uses the less accurate Modeller for comparative modeling in its standalone version. Sparks-X is very fast with good coverage, but the version available for download does not provide modeling, only template identification. I-Tasser is the slowest of all the tools considered, but it is very accurate and is ranked the best by CASP. Rosetta has good accuracy and coverage, but it requires a template protein and an alignment as an input defined by user. SWISS-MODEL is fast with good coverage, but it is not available as a standalone version. Modeller is one of the fastest and the most robust tools with reasonable accuracy for modeling cases with good templates. We selected two tools for implementation with HotSpot Wizard: (i) I-Tasser, which is ranked the most accurate of all the tools considered, but also very slow (∼3 days for an average-sized protein) and (ii) Modeller, which is less accurate, but very fast (∼5 min for an average-sized protein). Both tools can be run in a fully automatic mode, or the template protein and/or the pairwise alignment can be entered as an input information.

Quality assessment of the model

It is essential to assess the quality of the homology model prior to its further use for identification of hotspots or for the design of libraries. It is important to identify low quality models and the parts of the protein structure which were

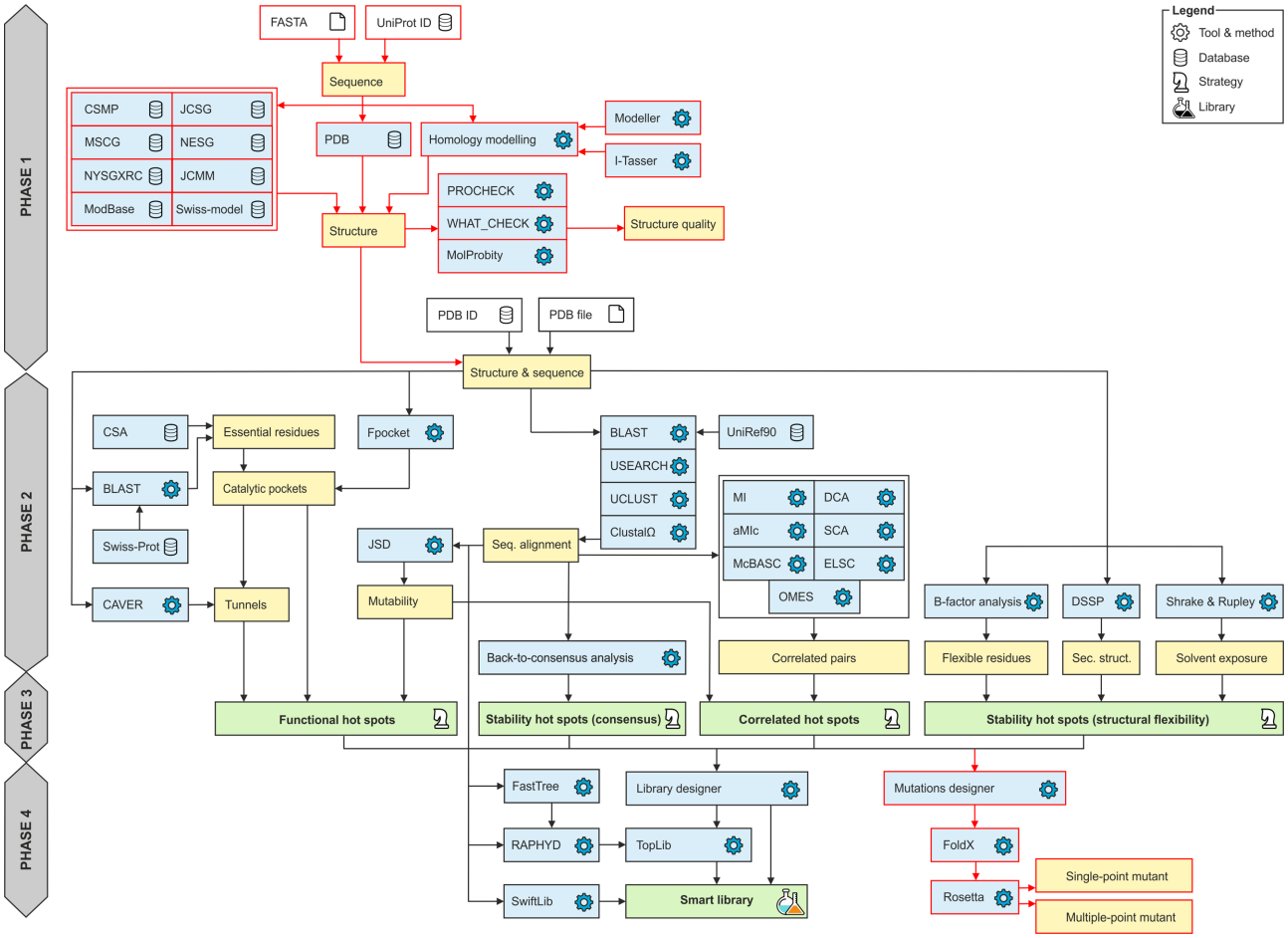

Figure 1. Workflow diagram of HotSpot Wizard 3.0. The workflow consists of four phases: (1) construction of a model of a structure, (2) annotation of a protein, (3) identification of mutagenesis hot spots and (4) design of mutations and a smart library. Phase 1 is applied only when a sequence is submitted as the input information. The new modules in version 3.0 are highlighted in red.

not modeled well. The results of today's modeling tools are far from perfect due to many difficulties with accurate protein structure prediction. Quality assessment is therefore an essential part of the phase 1 of the HotSpot Wizard workflow (Figure 1). Several quality assessment tools were considered and three of them, providing diverse quality metrics, were implemented. PROCHECK (40) is used for analysis of protein backbone torsion angles using Ramachandran diagrams and identification of the outliers from the allowed values. MolProbity (41) provides several parameters representing the quality of the whole structure as well as individual residues (number of poor rotamers, Ramachandran outliers, favored Ramachandran conformations, bad bonds and bad angles in the protein). WHAT_CHECK (42) generates a detailed report about structure quality (checks on secondary structure, coordinate problems, unexpected atoms, B-factor, occupancy checks, nomenclature related problems, geometric checks, torsion-related checks, bump checks, packing, accessibility, threading, water, ion and hydrogen bond-related checks).

Mutation design based on thermodynamic stability

Mutation design is part of the phase 4 of the HotSpot Wizard computation (Figure 1). Force field calculations are used for quantifying the change in protein thermodynamic stability after mutation. Rosetta (43) is used to evaluate $\Delta\Delta G$ between the wild-type and the mutant structures. Either single-point or multiple-point mutants can be evaluated. If the single-point mutations are pre-selected, multiple mutant structures are evaluated according to the user's selected positions and intended amino acid substitutions. The user can also select several mutations in a single round and calculate the energy of combined multiple-point mutants. For stability evaluation, FoldX (44) is first used for repairing protein structure by filling in the missing atoms and patching the structure. Then, minimalization of the structure using Rosetta is carried out using default settings. After that, a Rosetta stability calculation according to protocol 3 (45) is carried out, which results in the prediction of $\Delta\Delta G$ value for each mutation.

DESCRIPTION OF THE WEB SERVER

Sequence input and homology modeling

Initially, the user selects one of two types of input data: a structure or a sequence (Figure 2A). If a sequence is selected, there are three types of input. The user can either manually enter the protein sequence, specify the UniProt ID or upload the FASTA file. After entering the sequence, the user is provided with the results from searching the Protein Data Bank or the Protein Model Portal. This result is displayed in the form of a table (Figure 2B). In the case of the Protein Data Bank results, PDB ID, resolution and the link to the Protein Data Bank are provided. The user can then pick one of the proteins and continue with the HotSpot Wizard workflow. In the case of the results from the Protein Model Portal model provider, following information is listed: (i) used template, (ii) sequence identity with a template, (iii) range of the alignment, (iv) coverage and (v) reliability of the model. Links to a model in the Protein Model Portal and the template structure in the Protein Data Bank are provided in the table. Coverage and reliability of the models are represented by a color ranging from green to red (Figure 2C). If the user selects a model with unsatisfactory coverage (<80%) or insufficient reliability (low reliability value), a warning is displayed. When a protein model is selected which cannot be downloaded automatically, the user is asked to download it manually and then upload it as a structure for further analysis. The user can then select one of the models provided and continue with the HotSpot Wizard workflow or, if none of the models is satisfactory, carry out homology modeling and construct their own model. If the user carries out homology modeling, several parameters must be set first (Figure 2D). The user can select between Modeller, which is faster but less accurate, or I-Tasser, which is more accurate but slow. The second important parameter that must be specified prior to calculation is either automatic or manual identification of the template structure and alignment. The template can be provided either by entering the PDB ID or by uploading a PDB file. In the case of the user entering the alignment, pairwise alignment of the template and an input sequence in FASTA format must be provided. The process of hotspot identification can then begin after all these essential inputs have been defined.

Quality assessment of the model

Results of the quality assessment are shown in separate windows consisting of three tabs containing various quality assessment analyses. The first tab shows the MolProbity overall quality assessment table (Supplementary Figure S1A). In this table, the number and percentage of poor rotamers, Ramachandran outliers, favored Ramachandran conformers, bad bonds and bad angles are shown. Colored highlights are used to distinguish between good and unsatisfactory models. The second tab shows the MolProbity quality assessment results for each residue, displayed in the form of plots (Supplementary Figure S1B). A plot of MolProbity Ramachandran scores and MolProbity rotamer scores is given. In the last tab, there is a Ramachandran plot for the protein created by PROCHECK with outlier residues highlighted (Supplementary Figure S1C). The contents of all these tabs can be downloaded in PDF format together with a full quality assessment report created by WHAT_CHECK.

Mutations design based on stability

The stability changes introduced by specific mutations can be accessed through a newly introduced Mutations design module (Supplementary Figure S2A). There are three tabs in the Mutation design window—the first for definition of single-point mutants, the second for multiple-point mutants and the third summarizing the status of submitted jobs. In the case of single-point mutations, the user can select particular amino acids for each of the selected hotspots. The amino acid residues for mutagenesis can be selected based on: (i) amino acid frequency, (ii) mutational landscape, (iii) physico-chemical properties or (iv) user selection (Supplementary Figure S2B). After selection of the mutations, the stability of each single-point mutation is evaluated by the Rosetta software suite. The results are shown in the table—stabilizing mutations are highlighted in green, destabilizing mutations are highlighted in red (Supplementary Figure S2C). There are two options for setting multiple-point mutants. Either a particular amino acid can be selected for each position in the multiple-point tab or the results table from a previous single-point calculation can be used for recombination with the most promising substitutions. In both cases, only a single substitution for each position can be selected (Supplementary Figure S2D). After the calculation is finished, Hotspot Wizard reports the overall stability change as well as the decomposition of energy terms, both of which provide excellent assistance for mutagenesis experiments (Supplementary Figure S2E). The stability prediction can be downloaded in CSV format with the sequence of designed mutants being provided in FASTA format. These reports can also be generated in PDF or HTML formats. The third tab shows a table with the history of previously evaluated stabilities for the job. For each calculation, the job id, date and time of computation, status of the job (failed or finished), mutation type (single-point or multiple-point), selected positions and mutations are shown (Supplementary Figure S2F). The results page from any previous calculations can be revisited at any time.

EXPERIMENTAL VALIDATION

We have carried out validation of individual steps of the workflow as well as thoroughly tested the final version of the web server. The homology modeling tools were selected for implementation based on the results of CAMEO comparison (Supplementary Data 1). The reliability, coverage and availability of a standalone version of all the software code were considered during the selection process. The reliability of the Rosetta protocol 3 employed in the Design module was benchmarked against experimental stability data previously collected for multiple-point mutants in our laboratory (46) as well as 1573 single-point mutants available in the ProTherm and HotMuSiC databases (Supplementary Data 2). These tests confirmed a significant correlation between half-lives and calculated changes in free energy $\Delta\Delta G$, as well as an ability of the fast protocol 3 to correctly

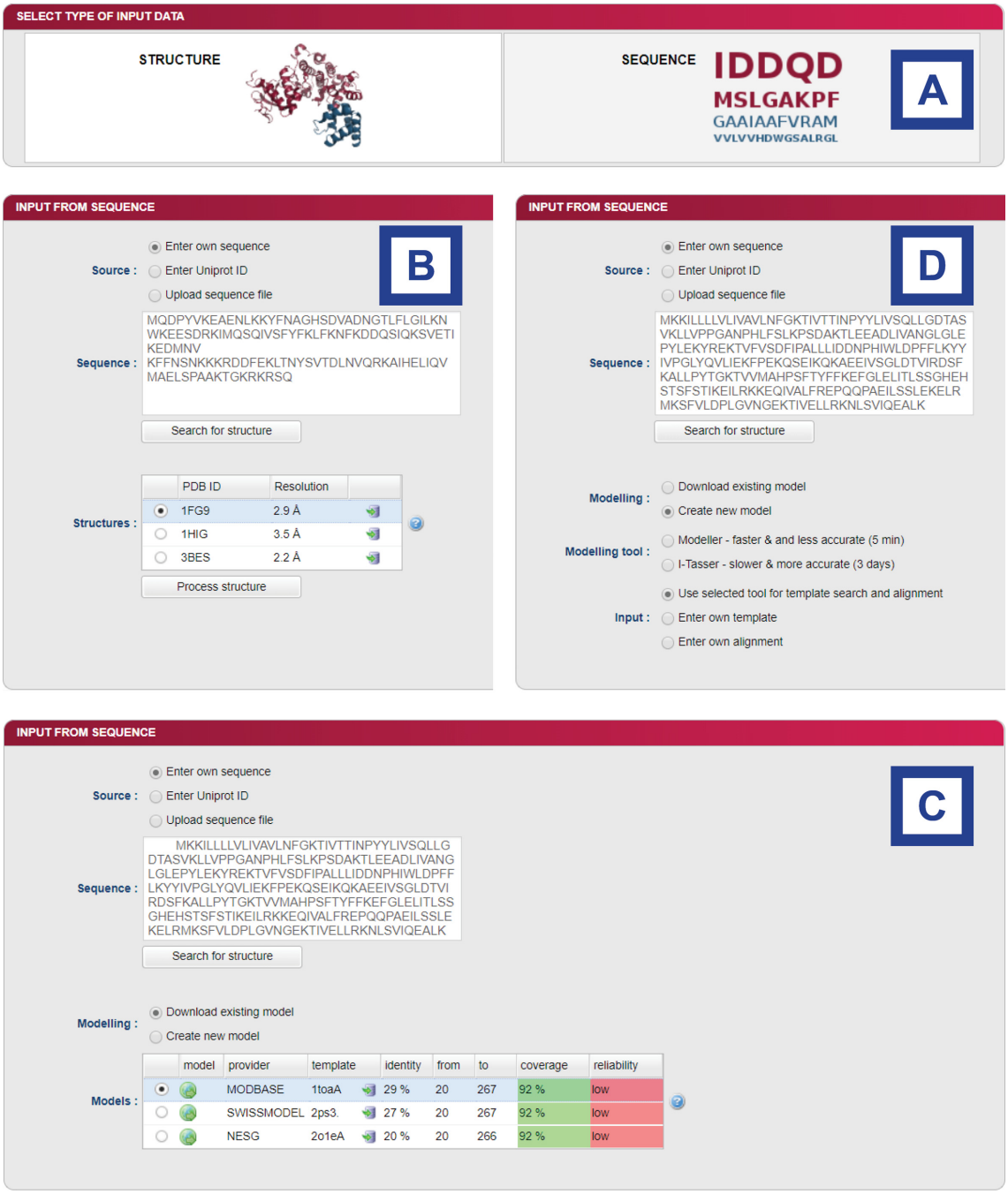

Figure 2. Graphic user interface of the sequence input in the HotSpot Wizard 3.0. (**A**) Selection between structure and sequence input. (**B**) After entering of the sequence, searching for existing structures in PDB database is performed. (**C**) If no existing structure is found, search in homology model databases is performed. (**D**) Setting of homology modeling parameters—user can choose between Modeller and I-Tasser and eventually enter his own template or sequence alignment.

classify stabilizing and destabilizing mutations. Functionality of the Mutation design module was validated by saturation mutagenesis at the hotspot position L177 located at the tunnel mouth of the haloalkane dehalogenase LinB (47). Theoretical predictions correctly identified the variant L177W, which was found to be the most stable also experimentally (Supplementary Data 3). At last, we used the HotSpot Wizard 3.0 workflow for computational mutagenesis of six residues lining the active site cavity and the access tunnel of the haloalkane dehalogenases from non-pathogenic and pathogenic bacteria *Sphingobium japonicum* UT26 and *Mycobacterium tuberculosis* Rv2579, respectively (48). Single-point mutations and combined sixfold mutants were predicted using the automated protocols with crystal structures and homology models (Supplementary Data 4).

CONCLUSIONS AND OUTLOOK

HotSpot Wizard 3.0 is a new version of a popular web server used for the automated prediction of hotspots and the design of smart libraries in semi-rational protein design. In this version, homology modeling of the protein structure dramatically increases the usability of the platform by increasing the number of possible inputs and solves the limitation imposed by the number of available experimental structures. For homology modeling, Modeller and I-Tasser are used. The quality of the models created is evaluated using three different tools to identify wrongly modeled regions, which should be used for further computational design only with extreme care. The users are automatically warned whenever they attempt to redesign poorly resolved regions, for example the residues lying outside allowed regions of the Ramachandran plot. Rational design is further supported by the novel Mutation design module employing force field calculations for estimating the effect of substitution on protein thermodynamic stability. This new module can dramatically reduce the number of variants selected for experimental testing and can also help to pre-select mutations for identified positions during construction of smart libraries. In the future, we want to focus on more systematic use of multiple structural data from the Protein Data Bank, and on development of a novel engineering strategy for the design of biocatalysts that catalyze specific chemical reactions. Extensive databases searches will be coupled with the computational design module for identification of the best starting protein template for such an engineering exercise.

SUPPLEMENTARY DATA

Supplementary Data are available at NAR Online.

ACKNOWLEDGEMENTS

Computational resources were supplied by the Ministry of Education, Youth and Sports of the Czech Republic under the Projects CESNET (Project No. LM2015042) and CERIT-Scientific Cloud (Project No. LM2015085).

FUNDING

Ministry of Education, Youth and Sports of the Czech Republic from the National Programme of Sustainability II [LQ1602, LQ1605, LO1214]; European Regional Development Fund [LM2015051, LM2015047, LM2015055]; Grant Agency of the Czech Republic [16–06096S]; European Union [720776, 722610]; Brno University Technology [FIT-S-17–3994 to L.S.]. Funding for open access charge: Czech Ministry of Education.
Conflict of interest statement. None declared.

REFERENCES

1. Hawkins,M.J., Soon-Shiong,P. and Desai,N. (2008) Protein nanoparticles as drug carriers in clinical medicine. *Adv. Drug Deliv. Rev.*, **60**, 876–885.
2. Godfrey,T. and Reichelt,J. (1982) Industrial applications. In: *Industrial Enzymology: The Application of Enzymes in Industry*. Macmillan, The Nature Press, London, pp. 582.
3. Bromley,E.H., Channon,K., Moutevelis,E. and Woolfson,D.N. (2008) Peptide and protein building blocks for synthetic biology: from programming biomolecules to self-organized biomolecular systems. *ACS Chem. Biol.*, **3**, 38–50.
4. De La Rica,R. and Matsui,H. (2010) Applications of peptide and protein-based materials in bionanotechnology. *Chem. Soc. Rev.*, **39**, 3499–3509.
5. Cheng,F., Zhu,L. and Schwaneberg,U. (2015) Directed evolution 2.0: improving and deciphering enzyme properties. *Chem. Commun.*, **51**, 9760–9772.
6. Romero,P.A. and Arnold,F.H. (2009) Exploring protein fitness landscapes by directed evolution. *Nat. Rev. Mol. Cell Biol.*, **10**, 866–876.
7. Lutz,S. (2010) Beyond directed evolution—semi-rational protein engineering and design. *Curr. Opin. Biotechnol.*, **21**, 734–743.
8. Cheng,Z., Peplowski,L., Cui,W., Xia,Y., Liu,Z., Zhang,J., Kobayashi,M. and Zhou,Z. (2017) Identification of key residues modulating the stereoselectivity of nitrile hydratase towards rac-Mandelonitrile by Semi-rational engineering. *Biotechnol. Bioeng.*, **115**, 1–12.
9. Bendl,J., Stourac,J., Sebestova,E., Vavra,O., Musil,M., Brezovsky,J. and Damborsky,J. (2016) HotSpot Wizard 2.0: automated design of site-specific mutations and smart libraries in protein engineering. *Nucleic Acids Res.*, **44**, W479–W487.
10. Talukdar,P. and Talapatra,S.N. (2017) Oxy-haemoglobin protein engineering: an automated design for hotspots stability, site-specific mutations and smart libraries by using HotSpot Wizard 2.0 software. *Int. J. Adv. Res. Comput. Sci.*, **8**, 220–228.
11. Wang,X., Ma,R., Xie,X., Liu,W., Tu,T., Zheng,F., You,S., Ge,J., Xie,H., Yao,B. *et al.* (2017) Thermostability improvement of a Talaromyces leycettanus xylanase by rational protein engineering. *Sci. Rep.*, **7**, 15287.
12. Vatansever,R., Uras,M.E., Sen,U., Ozyigit,I.I. and Filiz,E. (2016) Isolation of a transcription factor DREB1A gene from Phaseolus vulgaris and computational insights into its characterization: protein modeling, docking and mutagenesis. *J. Biomol. Struct. Dyn.* **35**, 1–12.
13. Berman,H.M., Westbrook,J., Feng,Z., Gilliland,G., Bhat,T.N., Weissig,H., Shindyalov,I.N. and Bourne,P.E. (2000) The Protein Data Bank. *Nucleic Acids Res.*, **28**, 235–242.
14. UniProt Consortium. (2017) UniProt: the universal protein knowledgebase. *Nucleic Acids Res.*, **45**, D158–D169.
15. Baker,D. and Sali,A. (2001) Protein structure prediction and structural genomics. *Science*, **294**, 93–96.
16. Cavasotto,C.N. and Phatak,S.S. (2009) Homology modeling in drug discovery: current trends and applications. *Drug Discov. Today*, **14**, 676–683.
17. Schwede,T. (2013) Protein modeling: what happened to the 'protein structure gap'? *Structure*, **21**, 1531–1540.
18. Haas,J., Roth,S., Arnold,K., Kiefer,F., Schmidt,T., Bordoli,L. and Schwede,T. (2013) The Protein Model Portal—a comprehensive resource for protein structure and model information. *Database*, **2013**, bat031.
19. Csmp.ucsf.edu. (2017) CSMP | Home. http://csmp.ucsf.edu/index.htm (20 December 2017, date last accessed).
20. Jcsg.org. (2017) The Joint Center for Structural Genomics (JCSG) Homepage. http://www.jcsg.org/ (20 December 2017, date last accessed).
21. Mcsg.anl.gov. (2017) http://www.mcsg.anl.gov/ (20 December 2017, date last accessed).
22. Nesg.org. (2017) NESG - NorthEast Structural Genomics consortium. http://www.nesg.org/ (20 December 2017, date last accessed).
23. Venkatagiriyappa,V. (2017) NYSGRC. http://www.nysgxrc.org/psi3-cgi/index.cgi (20 December 2017, date last accessed).
24. Jcmm.burnham.org. (2017) Joint Center for Molecular Modeling (JCMM). http://jcmm.burnham.org/ (20 December 2017, date last accessed).
25. Pieper,U., Webb,B.M., Dong,G.Q., Schneidman-Duhovny,D., Fan,H., Kim,S.J. and Tainer,J.A. (2013) ModBase, a database of annotated comparative protein structure models and associated resources. *Nucleic Acids Res.*, **42**, D336–D346.

26. Kiefer,F., Arnold,K., Künzli,M., Bordoli,L. and Schwede,T. (2008) The SWISS-MODEL repository and associated resources. *Nucleic Acids Res.*, **37**, D387–D392.
27. Biasini,M., Bienert,S., Waterhouse,A., Arnold,K., Studer,G., Schmidt,T. and Schwede,T. (2014) SWISS-MODEL: modelling protein tertiary and quaternary structure using evolutionary information. *Nucleic Acids Res.*, **42**, W252–W258.
28. Song,Y., DiMaio,F., Wang,R.Y.R., Kim,D., Miles,C., Brunette,T.J. and Baker,D. (2013) High-resolution comparative modeling with RosettaCM. *Structure*, **21**, 1735–1742.
29. Kim,D.E., Chivian,D. and Baker,D. (2004) Protein structure prediction and analysis using the Robetta server. *Nucleic Acids Res.*, **32**, W526–W531.
30. Kelley,L.A., Mezulis,S., Yates,C.M., Wass,M.N. and Sternberg,M.J. (2015) The Phyre2 web portal for protein modeling, prediction and analysis. *Nat. Protoc.*, **10**, 845–858.
31. Larsson,P., Skwark,M.J., Wallner,B. and Elofsson,A. (2010) Improved predictions by Pcons. net using multiple templates. *Bioinformatics*, **27**, 426–427.
32. Webb,B. and Sali,A. (2014) Protein structure modeling with MODELLER. *Methods Mol. Biol.*, **1137**, 151–115.
33. Yang,J., Yan,R., Roy,A., Xu,D., Poisson,J. and Zhang,Y. (2015) The I-TASSER Suite: protein structure and function prediction. *Nat. Methods*, **12**, 7–8.
34. McGuffin,L.J., Atkins,J.D., Salehe,B.R., Shuid,A.N. and Roche,D.B. (2015) IntFOLD: an integrated server for modelling protein structures and functions from amino acid sequences. *Nucleic Acids Res.*, **43**, W169–W173.
35. Russel,D., Lasker,K., Webb,B., Velázquez-Muriel,J., Tjioe,E., Schneidman-Duhovny,D. and Sali,A. (2012) Putting the pieces together: integrative modeling platform software for structure determination of macromolecular assemblies. *PLoS Biol.*, **10**, e1001244.
36. Hildebrand,A., Remmert,M., Biegert,A. and Söding,J. (2009) Fast and accurate automatic structure prediction with HHpred. *Proteins*, **77**, 128–132.
37. Källberg,M., Wang,H., Wang,S., Peng,J., Wang,Z., Lu,H. and Xu,J. (2012) Template-based protein structure modeling using the RaptorX web server. *Nat. Protoc.*, **7**, 1511–1522.
38. Yang,Y., Faraggi,E., Zhao,H. and Zhou,Y. (2011) Improving protein fold recognition and template-based modeling by employing probabilistic-based matching between predicted one-dimensional structural properties of query and corresponding native properties of templates. *Bioinformatics*, **27**, 2076–2082.
39. Kryshtafovych,A., Fidelis,K. and Moult,J. (2014) CASP10 results compared to those of previous CASP experiments. *Proteins*, **82**, 164–174.
40. Laskowski,R.A., MacArthur,M.W., Moss,D.S. and Thornton,J.M. (1993) PROCHECK: a program to check the stereochemical quality of protein structures. *J. Appl. Crystallogr.*, **26**, 283–291.
41. Chen,V.B., Arendall,W.B., Headd,J.J., Keedy,D.A., Immormino,R.M., Kapral,G.J. and Richardson,D.C. (2010) MolProbity: all-atom structure validation for macromolecular crystallography. *Acta Crystallogr. D Biol. Crystallogr.*, **66**, 12–21.
42. Hooft,R W., Vriend,G., Sander,C. and Abola,E.E. (1996) Errors in protein structures. *Nature*, **381**, 272–272.
43. Kellogg,E.H., Leaver-Fay,A. and Baker,D. (2011) Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins*, **79**, 830–838.
44. Schymkowitz,J., Borg,J., Stricher,F., Nys,R., Rousseau,F. and Serrano,L. (2005) The FoldX web server: an online force field. *Nucleic Acids Res.*, **33**, W382–W388.
45. Kellogg,E.H., Leaver-Fay,A. and Baker,D. (2011) Role of conformational sampling in computing mutation-induced changes in protein structure and stability. *Proteins*, **79**, 830–838.
46. Bednar,D., Beerens,K., Sebestova,E., Bendl,J., Khare,S., Chaloupkova,R., Prokop,Z., Brezovsky,J., Baker,D. and Damborsky,J. (2015) FireProt: energy-and evolution-based computational design of thermostable multiple-point mutants. *PLoS Comput. Biol.*, **11**, e1004556.
47. Chaloupková,R., Sykorova,J., Prokop,Z., Jesenska,A., Monincova,M., Pavlova,M., Tsuda,M., Nagata,Y. and Damborsky,J. (2003) Modification of activity and specificity of haloalkane dehalogenase from Sphingomonas paucimobilis UT26 by engineering of its entrance tunnel. *J. Biol. Chem.*, **278**, 52622–52628.
48. Nagata,Y., Prokop,Z., Marvanova,S., Sykorova,J., Monincova,M., Tsuda,M. and Damborsky,J. (2003) Reconstruction of mycobacterial dehalogenase Rv2579 by cumulative mutagenesis of haloalkane dehalogenase LinB. *Appl. Environ. Microbiol.*, **69**, 2349–2355.