UNIVERSITÀ DEGLI STUDI DI PADOVA

Sede Amministrativa: Università degli Studi di Padova
Dipartimento di : Biologia

CORSO DI DOTTORATO DI RICERCA IN: BIOSCIENZE E BIOTECNOLOGIE
CURRICOLO: BIOFISICA E BIOCHIMICA
CICLO: XXIX

SOLVING THE STRUCTURAL MODELING PROBLEMS FOR TANDEM REPEAT PROTEINS

Tesi redatta con il contributo finanziario della:
Commissione Europea, programma Erasmus Mundus Action 2 “Preciosa”
Pontificia Universidad Católica del Perú

Coordinatore: Ch.mo Prof. Paolo Bernardi
Supervisore: Ch.mo Prof. Silvio S. E. Tosatto
Co-Supervisore: Ph.D. Damiano Piovesan

Dottorando : Layla Hirsh Martinez

Abstract

Over the last decade, numerous studies have demonstrated fundamental importance of tandem repeat proteins (TRP) in many biological processes (Andrade, Perez-Iratxeta, and Ponting 2001). Repeat proteins are a widespread class of non-globular proteins carrying heterogeneous functions involved in several diseases. One of the most frequent problems in the study of biology is the functional characterization of a protein. This problem is usually solved by analyzing the three-dimensional (3D) structure. The experimental determination of the 3D structure is time consuming and technically difficult. For this reason structure prediction by homology modeling offers a fast alternative to experimental approaches. However homology modeling is not feasible for tandem repeat proteins because it is difficult to infer homology due to a high degree of sequence degeneration. In this thesis, I focused on algorithms oriented toward repeat unit prediction, and characterization. I developed an innovative approach, Repeat Protein Unit Predictor (ReUPred), for fast automatic prediction of repeat units and repeat classification, exploiting a Structure Repeat Unit Library (SRUL) derived from RepeatsDB, the core database of TRP. ReUPred is based on the Victor C++ library, an open source platform dedicated to protein structure manipulation. To prove the accuracy of the predictor, we ran it against all the entries in the PDB database and the resulting predictions allowed us to improve and increase RepeatsDB annotation twenty times. During my PhD I have integrated ReUPred prediction into the new version of RepeatsDB (release 2.0) that now features information on start and end positions for the repeat regions and units for all entries. The updated web interface includes a new search engine for complex queries and a fully re-designed entry page for a better overview of structural data. To further improve RepeatsDB quality we decided to provide a finer classification at the subclass level based on the structural conformation of the repeated units. We hypothesized that inside these ensembles it is possible to find subgroups of proteins sharing the same unit type. To prove it, we performed a detailed structural analysis. We created a network where nodes are the units and arcs represent structural similarity. The network can be partitioned in 7 different clusters. For each cluster, it was possible to create a Hidden Markov Model

similar to those representing Pfam domains. This analysis is an unpublished work but it already helped to improve ReUPred accuracy and RepeatsDB annotation. To summarize, this work is a partial answer to the problems of TRP modeling and might be helpful during future investigations such as drug design and disease studies.

Riassunto

Nell'ultima decade, numerosi studi hanno dimostrato il ruolo fondamentale svolto dalle proteine ripetute (TRP, tandem repeat proteins) in molti processi biologici (Andrade, Perez-Iratxeta, and Ponting 2001). Quella delle TRP è un'ampia classe di proteine non globulari, caratterizzate da una notevole eterogeneità di funzione e dall'essere coinvolte nella eziogenesi di numerose patologie. Una delle maggiori difficoltà che si incontrano nella moderna biologia è la caratterizzazione funzionale di proteine. Nella pratica standard, questo problema è affrontato analizzandone la struttura cristallografica (3D). Tuttavia, la determinazione della struttura tridimensionale è un processo molto lento e spesso inficiato da difficoltà tecniche. Per questa ragione, le tecniche computazionali di modellazione per omologia spesso offrono una alternativa praticabile all'approccio sperimentale. Tali tecniche però non sono di ausilio nello studio delle TRP. Ciò è dovuto all'impossibilità di poter inferire informazione evolutiva a causa di una ridotta conservazione di sequenza dell'unità ripetuta, a sua volta derivata da un elevato grado di degenerazione della sequenza primaria. In questo elaborato di tesi, mi sono focalizzata sullo sviluppo di un algoritmo orientato alla predizione di unità ripetute in proteine e alla loro caratterizzazione. Qui presento ReUPred (Repeat Protein Unit Predictor), un algoritmo innovativo per la predizione e caratterizzazione di unità proteiche ripetute basato sulla "libreria di unità strutturali ripetute" (SRUL, Structure Repeat Unit Library) direttamente derivata da RepeatsDB, la risorsa di riferimento per lo studio delle TRP. Architetturalmente, ReUPred è basato sulla libreria VICTOR C++, una piattaforma a sorgente aperto per la manipolazione di strutture proteiche. L'accuratezza del predittore è stata validata analizzando la banca dati PDB e le predizione ottenutene sono state successivamente utilizzate per estendere di venti volte il numero di proteine, correttamente annotate, contenute in RepeatDB. Durante lo svolgimento del mio dottorato ho integrato ReUPpred nella nuova versione di RepeatDB (release 2.0), che grazie a questo lavoro, ora integra informazioni dettagliate sulla posizione di inizio e fine per ogni unità ripetuta contenuta nel catalogo. L'interfaccia utente della banca dati è stata aggiornata implementando un nuovo motore di ricerca che

permette ora ricerche semantiche complesse. Inoltre, lo stile grafico delle singole schede è stato ridisegnato per una migliore visualizzazione dei dati strutturali. Al fine di migliorare ulteriormente la qualità dei dati contenuti in RepeatDB è stata fornita una classificazione più dettagliata delle unità strutturali ripetute, fino al livello di sottoclasse. Abbiamo ipotizzato che all'interno di questa raccolta di dati fosse possibile identificare sottogruppi di proteine condividenti la stessa unità strutturale di base. Una dettagliata analisi strutturale è stata condotta al fine di validare questa ipotesi. E' stata generata una rete in cui le singole unità ripetute vengono visualizzate come nodi interconnessi da archi che rappresentano la similarità strutturale. Ne è emerso che l'intero insieme può essere descritto da sette diversi raggruppamenti. Inspirati dalla rappresentazione dei domini proteici usata nella banca dati Pfam, per ognuno dei raggruppamenti è stato derivato un modello di Markov nascosto (Hidden Markov Model). Questa analisi, al momento in via di completamento, ha già permesso di migliorare l'accuratezza di ReUPred ed il livello di annotazione di RepeatsDB. In sintesi, questo lavoro fornisce una robusta base teorica per il futuro sviluppo di nuove tecniche per la predizione di struttura di TRP e può essere di grande aiuto per la comprensione dei meccanismi alla base di patologie umane e per lo sviluppo di nuovi approcci terapeutici.

Content table

List of Figures

List of Tables

1. Introduction

Proteins are amino acid molecules that are coded by our genes, form the basis of living tissue and play a central role in the biological processes. For example, catalysing reactions in our bodies, transporting molecules such as oxygen, keeping us healthy as part of the immune system and transmitting messages from cell to cell. Proteins are the building blocks of life and come in many different shapes and sizes. They are long chains of various combinations of amino acids. And a protein's shape is determined by its amino acid sequence. The distribution of nonpolar amino acids along a protein chain largely determines how the protein folds. (Raven 2014).

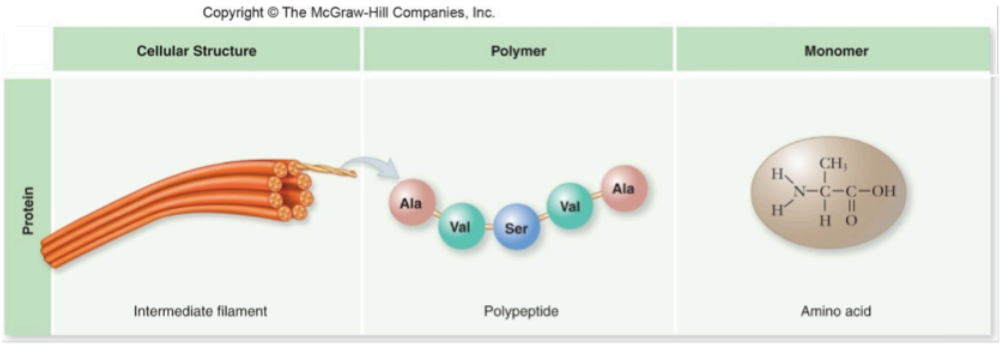


Figure 1 Representations of a protein (Raven 2014)

1.1. Protein structure

Proteins are three-dimensional objects; they are large, complex polymers. Their biological function is dictated by its three-dimensional structure. They are considered as "molecular workhorses" of the cell (*Privileged Scaffolds in Medicinal Chemistry: Design, Synthesis, Evaluation* 2016). Also they are a diverse class of biological polymers that play an extraordinary variety of functional roles. Proteins consist of long amino acid chains folded into complex shapes. X-ray diffraction is one of the methods available to solve atomic coordinates for protein structure determination and is a painstaking procedure that allows investigators to build up a three-dimensional image of each atom's position. The first protein to be analyzed in this way was myoglobin, soon followed by hemoglobin. As more and more proteins were added to the list, a general principle became evident: in every protein studied, essentially all the internal amino acids are nonpolar ones, as leucine, valine, and phenylalanine.

Water's tendency to hydrophobically exclude nonpolar molecules literally shoves the nonpolar portions of the amino acid chain into the protein's interior. This positions the nonpolar amino acids in close contact with one another, leaving little empty space inside. Polar and charged amino acids are restricted to the surface of the protein except for the few that play key functional roles. Protein structure has four structural levels that depend one on the other (Figure 2).

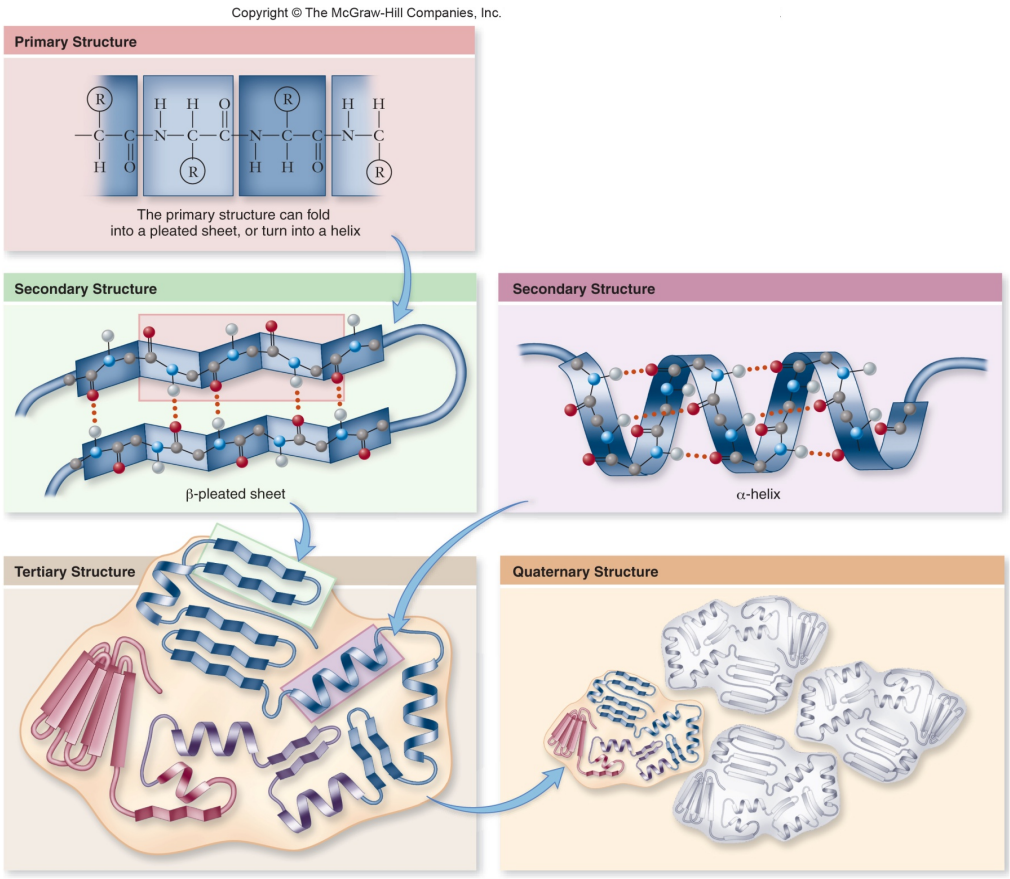


Figure 2 Levels of a protein structure (Raven 2014)

The first level is amino acid sequence and how it characterizes a single protein. The sequence of these amino acids in a polypeptide chain determines the types of secondary structure elements; the folding of the amino acid chain by hydrogen bonding into coils and pleats is called a protein's secondary structure and it is the secondary level. The third level of organization is the way in which the secondary structure is arranged in space (motifs, folds and domains). Finally the fourth level is the quaternary structure, consisted of several polypeptide chains embedded in a protein complex. Because of progress in our

knowledge of protein structure, two additional levels of structure are increasingly distinguished by molecular biologists: motifs and domains. To summarize, in general; protein structure can be viewed at six levels: 1. the amino acid sequence, or primary structure; 2. coils and sheets, secondary structure; 3. folds or creases, called motifs; 4. three-dimensional shape, tertiary structure; 5. functional units, called domains; and 6. individual polypeptide subunits associated in a quaternary structure (Raven 2014).

1.2. Amino Acids

The amino acids are the building blocks of proteins; about 23 of them have been isolated from natural proteins. They have a specific characteristic defined by its side chain, which provides it with a unique role in a protein structure.
Each amino acid consists of an α carbon atom to which is attached (Figure 3):

- A hydrogen atom.
- An amino group (hence "amino" acid).
- A carboxyl group (-COOH). This gives up a proton and is thus an acid (hence amino "acid").
- One of 20 different "R" groups. It is the structure of the R group that determines which of the 20 it is and its special properties.

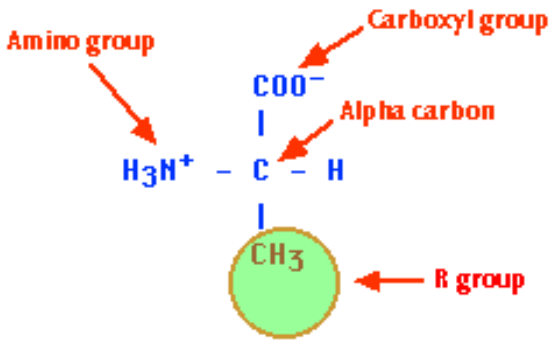


Figure 3 Alanine, its amino acid composition

Amino acids are linked into a polypeptide chain on the ribosome during protein synthesis. One possible classification is based on the propensity of the side chain to be in contact with polar solvent (water) as:

- Hydrophobic (low propensity to be in contact with water)
- Polar or charged (energetically favorable contact with water).

The charged amino acid residues include lysine (+), arginine (+), aspartate (-) and glutamate (-). Polar amino acids include serine, threonine, asparagine, glutamine, histidine and tyrosine. The hydrophobic amino acids include alanine, valine, leucine, isoleucine, proline, phenylalanine, tryptophan, cysteine and methionine (Table 1). Glycine on the other hand does not have a side chain, this is why it is not straightforward to assign it as hydrophobic or polar. It is also one of the most common amino acids. Generally, glycine is often found at the surface of proteins, within loop or coil regions, which gives to this location a high flexibility to the polypeptide chain. This suggests that it is rather hydrophilic. While proline, is usually found buried inside the protein, it is considered as non-polar and it is often found in loop regions.

Table 1 20 most common amino acids in protein

Charged (side chains often make salt bridges)		
Arginine	Arg	R
Lysine	Lys	K
Aspartic acid	Asp	D
Glutamic acid	Glu	E
Polar (usually participate in hydrogen bonds as proton donors or acceptors):		
Glutamine	Gln	Q
Asparagine	Asn	N
Histidine	His	H
Serine	Ser	S
Threonine	Thr	T
Tyrosine	Tyr	Y
Cysteine	Cys	C
Methionine	Met	M
Tryptophan	Trp	
Hydrophobic (normally buried inside the protein core):		
Alanine	Ala	A
Isoleucine	Ile	I
Leucine	Leu	L
Phenylalanine	Phe	F
Valine	Val	V
Proline	Pro	P
Glycine	Gly	G

In contrast to glycine, proline provides rigidity to the polypeptide chain by imposing certain torsion angles on structure segments. Proline in contrast to glycine fixes torsion angles at a certain value, very close to that of an extended

β-strand. It is usually found at the end of helices and functions as a “helix disruptor”. As Glycine and proline are essential for the conservation of a particular protein fold they are usually conserved within a protein family (Raven 2014) .

Most protein molecules have a hydrophobic core not accessible to solvent and a polar surface in contact with the environment. While hydrophobic amino acid residues build up the core, polar and charged amino acids preferentially cover the surface of molecules and are in contact with solvents due to their ability to form hydrogen bonds.

1.3. Primary structure

Proteins are made up of polypeptide chains, which are amino acids joined together with peptide bonds (Figure 4). The unique sequence of amino acids that make up a protein or polypeptide chain is called Primary Structure. It is a structure of a biological molecule in which there is a precise sequence or order of monomeric units. It serves as the covalent backbone of biological molecules (such as DNA and proteins) (Alberts 2002).

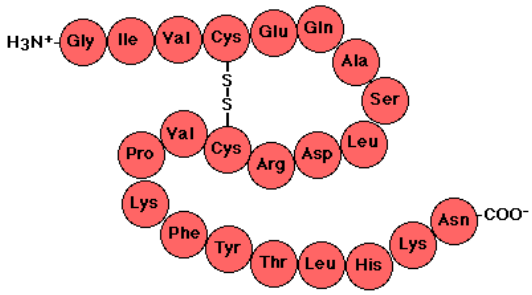


Figure 4 Graphical representation of a primary structure

1.4. Torsion angles

Torsion angles are dihedral angles, which are defined by 4 points in space. In proteins two torsions angles φ and ψ describe the rotation of the polypeptide chain around two bonds on both sides of the Cα atom (Figure 5). The Ramachandran plot (Figure **6**) is a way to view the distribution of torsion angles in a protein structure. It also shows excluded regions in which rotations of

polypeptide are not allowed due to collisions between atoms (steric hindrance).

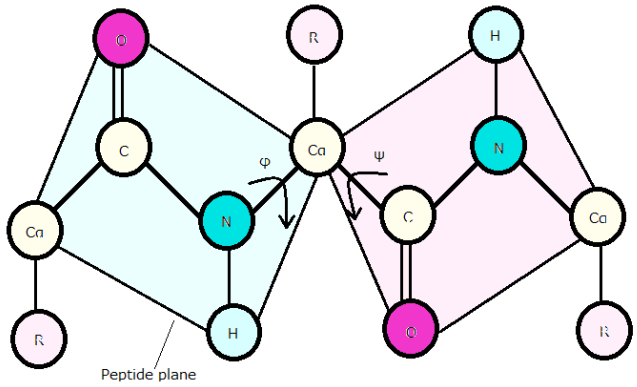


Figure 5 Rotations of the polypeptide backbone

This kind of plot of a particular protein serves also as an indicator of the 3D structure quality. Torsion angles are really important local structural parameters that control protein folding. The torsion angles phi and psi provide flexibility required for polypeptide backbone to adopt a certain fold, while omega (ω) is essentially flat and fixed to 180 degrees (Figure **6**) (*Metalloproteins: Structural Aspects* 1991).

Due to the partial double-bond character of the peptide bond, which restricts rotations around the C-N bond, placing two successive α-carbons and C, O, N and H between them in one plane. Thus, rotation of protein chain can be described as rotation of peptide bond planes relative to each other (Alberts 2002).

As shown in Figure **6** each type of secondary structure elements occupies its characteristic range of φ and ψ angles, marked α is for α-helices and β is for β-sheet on the left. Red indicates low-energy regions; brown allowed regions, yellow the so-called generously allowed regions and pale-yellow marks disallowed regions. On the left plot there are many dots in the disallowed regions, but almost none on the right (the ones which are seen are for glycine residues). The torsion angles on the left plot lack real clustering around secondary structure regions and have a much wider distribution, compared to the plot on the right. Generally this is a result of bad geometry - high resolution structures generally tend to have better clustering within the allowed regions of the plot (Anfinsen et al. 1981). The horizontal axis shows φ values, while the

vertical shows ψ values. Each dot on the plot shows the angles for an amino acid. Regions on Ramachandran plot with highest density of dots are called “allowed” or low-energy regions. Some values of φ and ψ are forbidden since involved atoms will come too close to each other, resulting in a steric clash. For a high-quality and high-resolution experimental structure these regions are usually empty or almost empty - very few amino acid residues in proteins have their torsion angles within these regions.

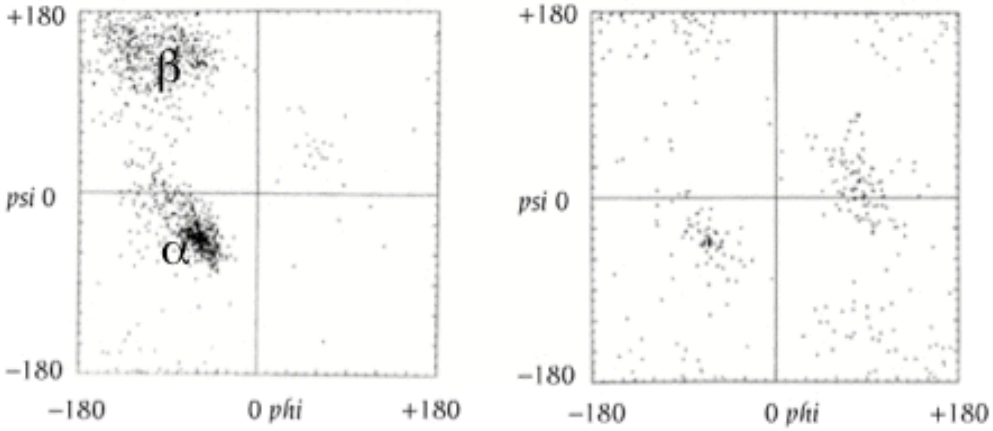


Figure 6 Ramachandran plot

In Ramachandran plot (Figure 7) for glycine an exception of clustering principle around the α and β-regions can be seen, as glycine does not have a side chain, which gives high flexibility to polypeptide chain, some of the forbidden rotation angles became accessible.

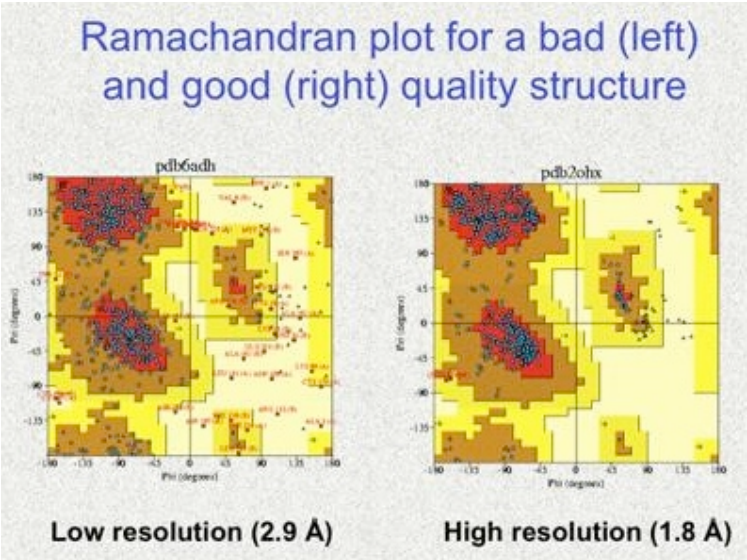


Figure 7 Two Ramachandran plots for the same structure refined at different resolutions.

This type of plot is also used in assessing quality of experimental structures or homology models, as torsion angles outside low-energy regions may indicate problems in structure, but they may also be true and may provide some interesting insights into the function of proteins.

1.5. Secondary structure

The –COOH and –NH2 groups of the main chain together with the amino acid side group can form hydrogen bonds. In the case of the main chain groups, the formed bonds could be as good that their interactions with water. It might be expected to offset the tendency of nonpolar side groups to be forced into the protein interior.

The reason for this is that the polar groups of the main chain form a kind of bonds with each other. As a result two patterns of H bonding occur. The first, in where hydrogen bonds form along a single chain, linking one amino acid to another farther down the chain, which tends to pull the chain into a coil (Figure 8). In the second pattern, the bond occurs across two chains, linking the amino acid of one chain to the other in the second chain. Usually many parallel chains are linked, forming a pleated, sheet-like structure (Figure 9) This two characteristic coils and pleats are the secondary structure (Raven 2014).

1.5.1. α Helix conformation

The R groups of amino acids all extend to the outside. The helix makes a complete turn every 3.6 amino acids. It is right-handed; and twist in a clockwise direction. The carbonyl group (-C=O) of each peptide bond extends parallel to the axis of the helix and points directly at the –N-H group of the peptide bond 4 amino acis below it in the helix. A hydrogen bond forms between them [-N-H…..)=C-] (Figure 8) (Raven 2014).

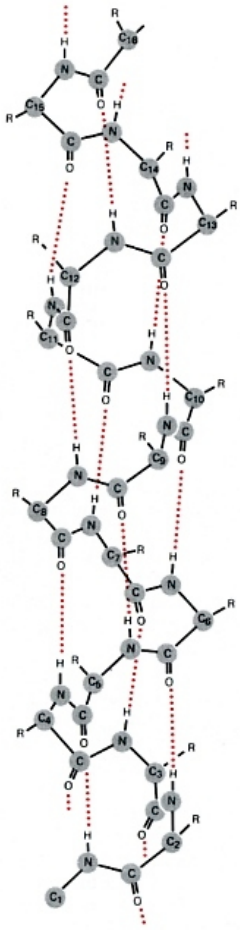


Figure 8 α hélix conformation

1.5.2. β-sheet Conformation

Consists of pairs of chains lying side-by-side and stabilized by hydrogen bonds between carbonyl oxygen atom on one chain and -NH group on the adjacent chain. The chains are often "anti-parallel"; N-terminal to C-terminal direction of one being the reverse of the other (Figure 9) (Raven 2014).

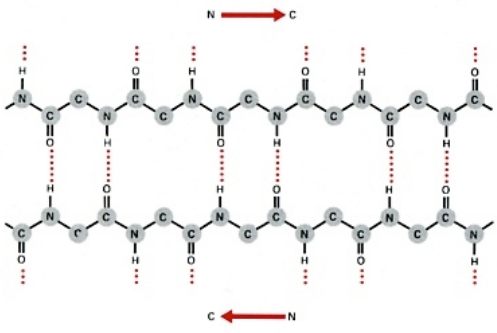


Figure 9 β pleated sheet

1.6. Tertiary structure

It refers to three-dimensional structure of the entire polypeptide chain in the native state, i.e. the most stable in physiological conditions. The final folded shape of a globular protein, which positions the folds and motifs non polar side groups into the interior is called a proteins tertiary structure (Raven 2014). Amino acid side chains may interact and bond in a number of ways. Interactions and bonds of side chains within a particular protein determine its tertiary structure and it is defined by it atomic coordinates. Many proteins can be fully unfolded ("denatured") and will spontaneously refold back to their characteristic shape. The action of heat can break a tertiary structure because with the increase of kinetic energy the structure vibrates more so the bonds that maintain its shape are more likely to break (denatured) (Figure 10). As the function of a protein depends on structure if denatured the function might be lost too. An example of a lost function are enzymes that when denatured they lose their catalytic power, or antibodies that can no longer bind to an antigen. If a mutation happens in the gene encoding of a protein, usually the tertiary structure is altered (Andersen 2001).

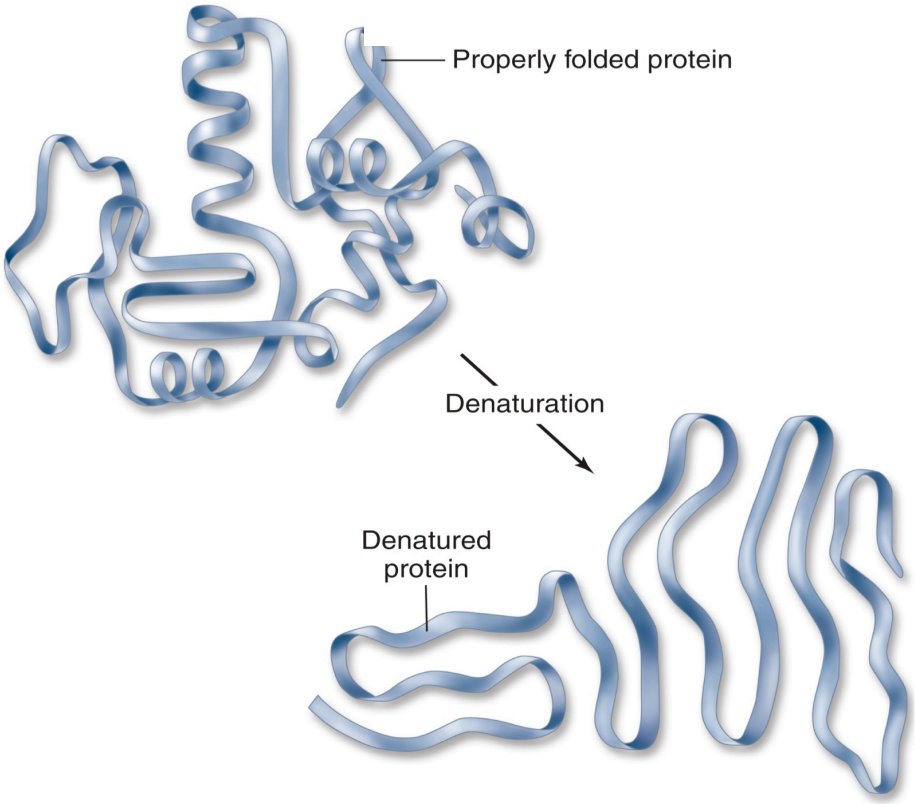


Figure 10 Denaturation of a protein(Raven 2014)

The stability of a protein is influence by how well its interior fits together. When two nonpolar chains in the interior are in very close proximity, they experience a kind of molecular attraction called Van der Waal's forces. By their own they are weak but when many come into play these forces can add up to a string attraction but they are effective only over short distances. As there are many different nonpolar amino acids with different-sized R groups, there are many precise fitting of non-polar chains within the protein interior. This is the reason why when mutation converts one nonpolar amino acid into another the protein's stability very often is disrupted and can result in lost or altered function of the protein. This tertiary structure may involve coiling or pleating, often with straight chains of amino acids in between.

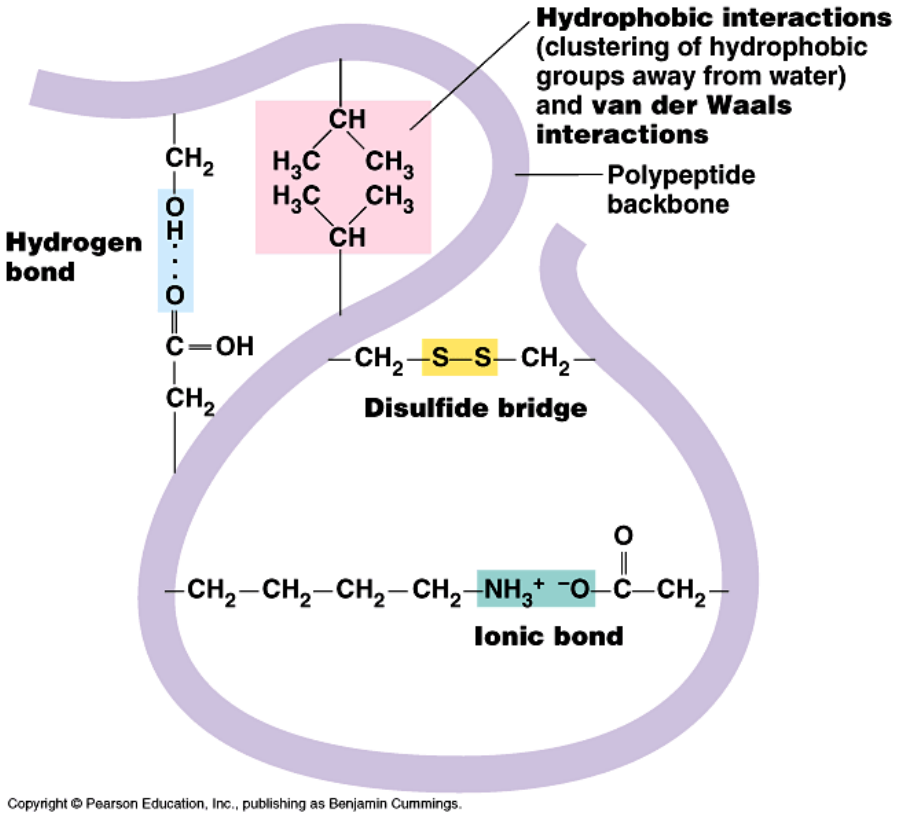


Figure 11 3D structure representation

Tertiary structure is held together by four different bonds and interactions:

- Disulphide Bonds Disulphide Bonds - Where two Cysteine amino acids are found together, a strong double bond (S=S) is formed between Sulphur atoms within Cysteine monomers.
- Ionic Bonds - If two oppositely charged 'R' groups (+ve and -ve) are found

close to each other, and ionic bond forms between them.

- Hydrogen Bonds - Your typical everyday Hydrogen bonds.
- Hydrophobic and Hydrophilic Interactions - Some amino acids may be hydrophobic while others are hydrophilic. In a water based environment, a globular protein will orientate itself such that it's hydrophobic parts are towards its centre and its hydrophilic parts are towards its edges (Figure 11).

Based on solubility there are two main groups (Alberts 2002):

- Globular – these kind of structure tend to form ball-like, its hydrophobic parts are towards the center and hydrophilic are towards the edges, which makes them water-soluble. Usually they have metabolic roles, for example: enzymes in all organisms, plasma proteins and antibodies in mammals.
- Fibrous – mostly consist of repeated sequences of amino acids, they are insoluble in water and form long fibers, usually have structural roles, such as: Collagen in bone and cartilage, Keratin in fingernails and hair.

Intrinsically disordered regions and proteins show a wide variety of structural subtypes. These different types of disorder can be characterized using an array of experimental techniques, and several resources collect computationally identified and experimentally verified disordered regions. Proteins have been proposed to function within a conformational continuum, ranging from fully structured to completely disordered (van der Lee et al. 2014).

1.7. Structural Motifs & Connectivity

The elements of secondary structure can combine in characteristic ways called motif or “supersecondary structure”. A motif is a short protein set of amino acids that contributes to the biological function of sequence in which it resides. A protein sequence *motif* is a short pattern that is conserved by purifying selection and may correspond to a protein binding site; in proteins, a motif may correspond to the active site of an enzyme or a structural unit necessary for proper folding of protein. Thus, sequence motifs are one of the basic functional units of molecular evolution. Consequently, identifying and understanding these

motifs is fundamental to building models of cellular processes at molecular scale and to understanding the mechanisms of human disease(Grant, Bailey, and Noble 2011).

One very common motif is the β α β motif, which created a fold (Figure 12). The "Rossmann fold", is another example of motif but a β α β α β one. Another motif that occurs in many proteins is the β-barrel, a β sheet folded round to form a tube. There is another really important motif that many proteins use to bind the DNA double helixes, that is the α turn α motif (Figure 12). They are short segments of protein 3D structure, which are spatially close but not necessarily adjacent in the sequence. An example of motif is the β turn is a structural motif with a structural role and it consists of four consecutive residues where the polypeptide chain folds back on itself by nearly 180 degrees(Raven 2014).

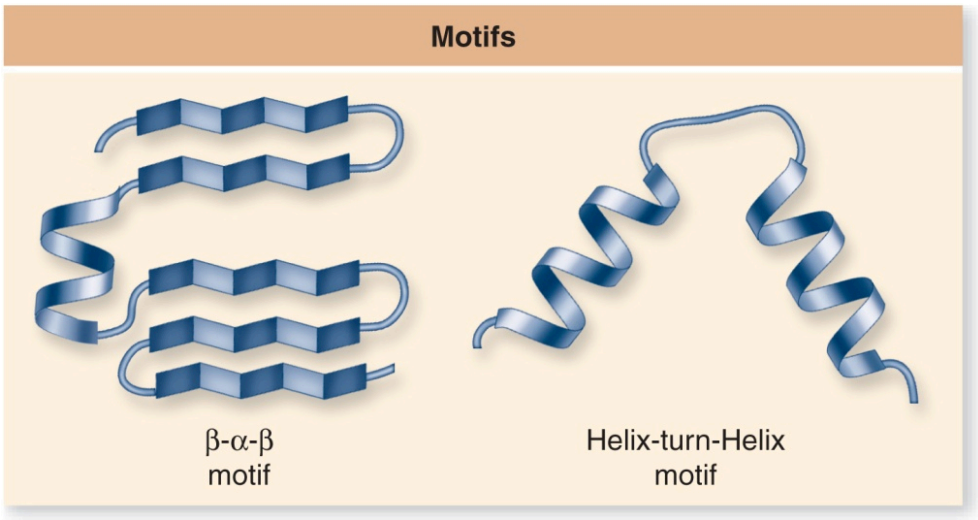


Figure 12 Structural Motif(Raven 2014)

1.8. Protein structure databases

The restrictions that nature places on three-dimensional structures during evolution are much stricter than those that it puts on amino acid sequence. The 3D structures generated by crystalografic and spectroscopic studies are limited but their impact will increase as more structures become available. In general, these resources can be divided into those that house the 3D coordinates of solved structures and those that classify and summarize them.

The PDB databank, is the single largest global archive of biological

macromolecular structures (Berman et al. 2000). It is a massively redundant resource because particular proteins become the focus of repeated structure determination. The nature of the information presented by structural classification schemes is dependent on the methods used to identify and evaluate similarity. Two well-known structure classification resources are SCOP and CATH.

SCOP (Structural Classification of Proteins) database classifies proteins of known structure according to their evolutionary and structural relationships (Andreeva et al. 2008). Domains in SCOP are grouped by species and hierarchically classified into families, superfamilies, folds and classes. This database has been constructed using a combination of manual and automated methods (Higgs and Attwood 2005) (Cuff et al. 2011) (Andreeva et al. 2008) and it is divided in.

- Family:

  Inside a family we can identify a clear evolutionary relationship. Usually this means that between proteins there is a pairwise residue identity greater or equal than 30%. However there are some cases where globins have only 15% sequence identity even though they share a common descent.
- Super family:

  In this level we can find proteins with low sequence identity with a probable common evolutionary origin suggested by its structural and functional features. An example for this level is actin, ATPase domain of heat shock protein, and hexakinase that together form a superfamily.
- Fold:

  When proteins have the same topological connections and same secondary structure we could say that they have a common fold. Proteins with different folds with different size and conformations often have peripheral elements of secondary structure and turn regions. Proteins with same fold category may not have a common evolutionary origin but similarities in structure may arise from physics and chemistry of proteins.

Another important database is CATH, that comes from the first letters in Class-Architecture-Topology-Homologous. It is a hierarchical domain classification of protein structures (Pearl et al. 2005). The resource is largely derived from

automatic methods, but when they fail manual inspection is used. There are four levels (Higgs and Attwood 2005):

- Class:
  Denotes a gross secondary structure content and packing
- Architecture:
  Describes a gross arrangement of secondary structures ignoring connectivity and assigned manually using simple descriptions, such as barrel, roll, sandwich.
- Topology:
  Assigns the overall shape and secondary structure connectivity by means of structure comparison algorithms, structures in which at least 60% of the larger protein matches the smaller are assigned to the same level.
- Homology:
  Clusters domains that share greater than or equal to 35% sequence identity and are thought to share a common ancestor. Both sequence and structure comparison algorithms identify similarities.

1.9. Protein domains

Proteins in our body are encoded within our genes in functional sections called exons. Each exon encodes a 100 to 200 residues section of a protein, and folds into a structurally independent functional unit call domain.

A domain (Figure 13) is the basic building block of a protein structure. Their main features are (Raven 2014) (Higgs and Attwood 2005):

a) It is a spatially separated unit of the protein structure.
b) It may have sequence and/or structural resemblance to another protein structure or domain.
c) It may have a specific function associated.

Certain protein domains have some clearly defined function associated with them, like Rossmann-fold domain, also called coenzyme-binding domain. Such domains often “carry” their function with them when they get inserted into different proteins during evolution.

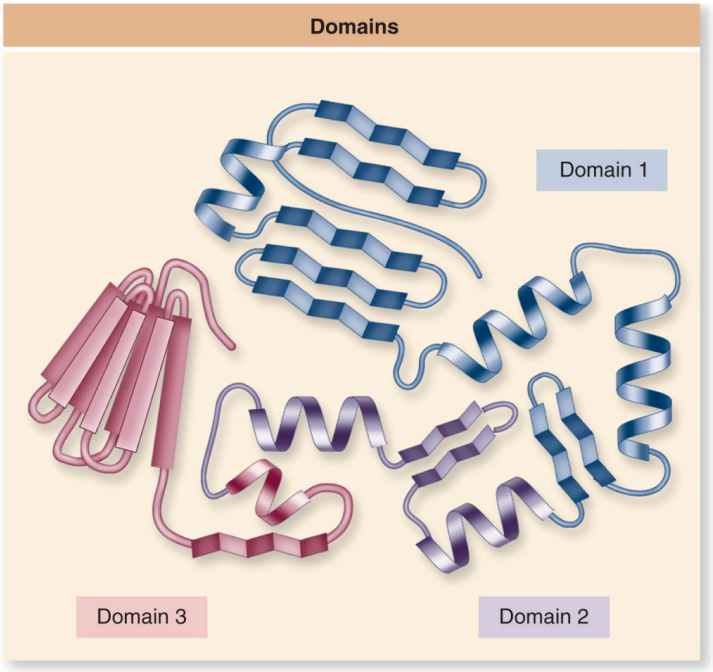


Figure 13 Domain representation (Raven 2014)

1.10. Thesis objectives

This document is a summary of all my research done since January 2014 which includes three published papers: the first published in Bioinformatics Journal (L. Hirsh et al. 2015), the second in Amino Acids ((L. Hirsh et al. 2016)) and the last one published in the Databases Special Issue of Nucleic Acid Research (Paladin et al. 2016) and also includes an already accepted but unpublished work that will be published in the Repeat Proteins Special Issue of Journal of Structural Biology.

In January 2014 I started my research trying to do homology modeling of repeat proteins, but in time I realized not only that there was not enough quality data to do it but also that homology modeling based on one template protein was not the best method to create a model of a repeat one.

Doing an analysis on the posible methods we could use for the modeling, we decided that the best way would be to work with repeat unit fragments as templates instead of a complete protein and then follow a homology model method. To do so we created Victor C++ library for protein representation and advance manipulation(L. Hirsh et al. 2015), the idea with this library was to have all what we needed to do the model once we have the template created

with the repeat unit fragments. The biggest problem was that at that time there was not enough data of the unit fragments which lead my research on a “different” path, the identification of repetitive units in protein structures with ReUPRed (L. Hirsh et al. 2016). The work we published was done to identify repeat unit fragments in solenoid proteins, but as the method accuracy was really high we replicated the work in elongated structures (Class III), closed structures (Class IV) and beads on string (Class V). The results were really interesting and let us identify new structural subclases. In paralel we had RepeatsDB (Di Domenico et al. 2014a), a database created in BiocomputingUp Laboratory (laboratory in where I have been doing my research) and all the data predicted by ReUPred, therefore we decided to make an update of the database , but to do so we manually curated some of the predicted information. We created RepeatsDB 2.0; with improved annotation classification, search and visualization of repeat protein structures (Paladin et al. 2016). In this database we present new structural subclasses while in the first release of the database only ˜300 entries had repeat unit information. In the new release all the entries have it and almost 60% of the entries (˜3500) were curated manually.

Finally, during the curating process we realized that each subclass repeat unit presented specific caracteristics inside them. We decided to do a deeper analysis and for that we started with the α-solenoid. This analysis is not yet published but the results lead us to believe that we could replicate the analysis in the rest of the classes and that we will be able to classify a protein based on its sequence. With this knowledge we could optimize the creation of the model, but this will be done in future research.

2. Tandem Repeat Proteins (TRP)

Tandem repeats (TRs) are ubiquitous, unstable genomic elements that have historically been designated as nonfunctional "junk DNA" and are therefore mostly ignored in comparative genomics. However, as many as 10% to 20% of eukaryotic genes and promoters contain an unstable repeat tract. Mutations in these repeats often have fascinating phenotypic consequences. For example, changes in unstable repeats located in/or near human genes can lead to neurodegenerative diseases such as Huntington disease. Apart from their role in disease, variable repeats also confer useful phenotypic variability, including cell surface variability, plasticity in skeletal morphology, and tuning of circadian rhythm. As such, TRs combine characteristics of genetic and epigenetic changes that may facilitate organismal evolvability (Gemayel et al. 2010).

More recent measurements (Pellegrini, Renda, and Vecchio 2012) give a count of about 20% of proteins in UniProtKB database (Bairoch 2004) holding a TRP of at least 20 amino acids length. In the last few years the number of known structures has been growing exponentially and in fact current (January 2017) PDB database holds 126,060 protein structure entries while in 1998 it only contained 2058 structure entries and in 2012, when Pellegrini et al. did their analysis, it had about 8775 structures entries. It is also important to notice that in all these years new structures have been found and new repeat structures have been identified and along the years we can observe an exponential growth.

Andrade, Perez-Iratxeta, and Ponting 2001 observed that repetitive subsequences that appear in tandem repetitions (TR) within protein primary sequence often form integrated assemblies when these residues are mapped to their corresponding 3D folded conformation. Tandem repeats mean different binding opportunities and may play a structural role by giving rigidity to a protein. Furthermore, repeats in protein sequences are usually hard to detect because on average repeating unit is relatively short, and moreover there can be considerable sequence divergence among units of the same TR (Andrade et al. 2001).

Some proteins show tandem repetitions of apparent modular structure that do not fold independently, but rather co-operate in stabilizing structural forms that

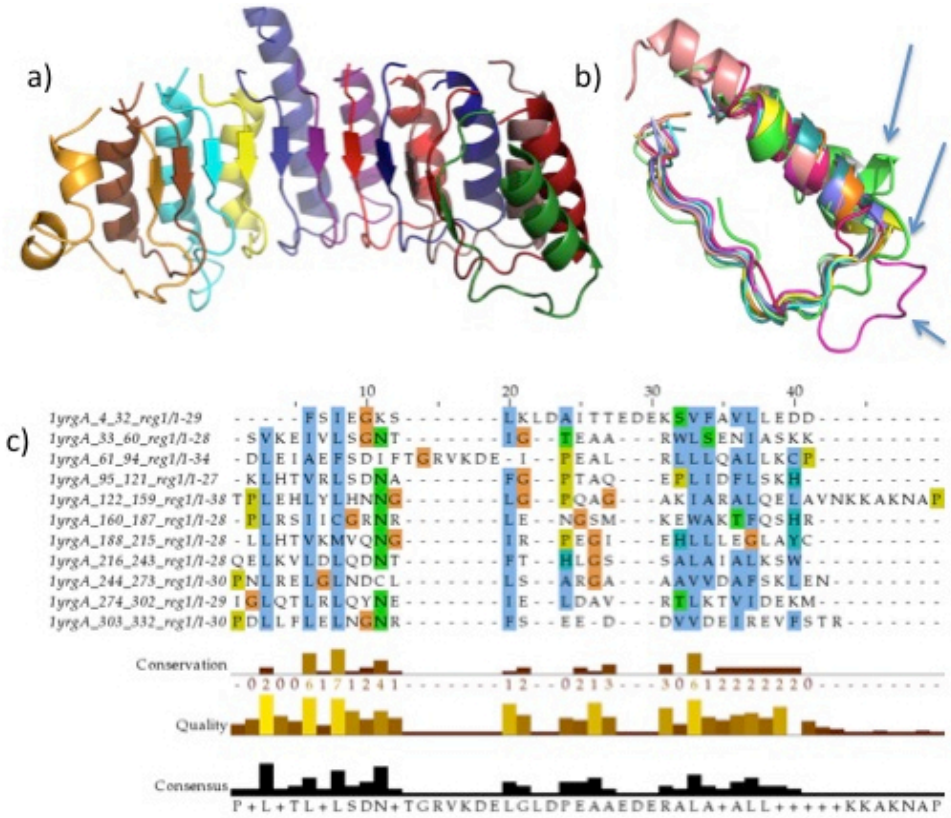
comprise several repeat-units (Espada et al. 2015). Some of these called protein domains are composed of units of similar structure. Often, but not always, these units are also similar in sequence. These kinds of domains can be considered repeats that originated by duplications from a single ancestral sequence. These small units are large enough to form secondary structural elements but too small to be stable by themselves. They acquire stability by folding together in a repetitive structure. Detection of TRP either from protein sequence or structure data is challenging due to inherent high signal to noise ratio (Pellegrini 2015).

Repeats, as mentioned before, are ubiquitous elements of proteins and play important roles for cellular function and during evolution. Repeats are also notoriously difficult to capture computationally and large scale studies so far had difficulties in linking genetic causes of diseases, structural properties and evolutionary trajectories of protein repeats. Shuler and Bornber-Bauer observed that repeats in larger protein families experience generally very few insertions or deletions (indels) of repeat units but there is also a significant fraction of noteworthy volatile outliers with very high indel rates (Schüler and Bornberg-Bauer 2016). Their analysis of structural data indicates that repeats with an open structure and independently folding units are more volatile and more likely to be intrinsically disordered. Such disordered repeats are also significantly enriched in sites with a high functional potential such as linear motifs (short stretches of protein sequence that mediate protein – protein interaction). In addition, the most volatile repeats have a high sequence similarity between their units. Since many volatile repeats also show signs of recombination, they conclude that these repeats are often shaped by concerted evolution, that is a molecular process that leads to homogenization of DNA sequences belonging to a given repetitive family (Liao 1999). Intriguingly, many of these conserved yet volatile repeats are involved in host-pathogen interactions where they might foster fast but subtle adaptation in biological arms races.

Quite often, these domains are reused across many proteins in a different context, i.e., they combine with other domains in changing order and quantity (Apic, Gough, and Teichmann 2001). Some protein repeats are hyper-variable, i.e., the number of repeat units within the domain repeat changes within evolutionary short time by insertion (expansion) or deletion (contraction) of

units. In Figure 14 we can observe 1yrg chain A protein, an αβ-solenoid in where a) each of the repeated units is colored b) has all the colored units structurally aligned, blue arrows show the biggest insertions c) shows the corresponding multisequence alignment of b) and where exactly the insertions are. In some proteins, these changes occur so rapidly that they can be observed not only between species but also within a population under both adaptive(Verstrepen et al. 2005) and neural conditions (Chevanne et al. 2010).

Figure 14



Tandem repeat protein
1YRG chain A a) 3D structure with each repeat unit colored b) All the units structurally aligned, marked with arrows three of the biggest insertions c) Corresponding sequence alignment of all the repeat units inside the chain, in squares the three biggest insertions.

2.1. Importance of TRP

There is a lot of theoretical interest in tandem repeat proteins because of their abundance, raising the question what is their role in evolution and in the structure of the genome. Other important questions include how do they appear, how do they evolve and why they do it.

Something we can assertate is that TRPs are important not only because of their involvement in diseases, but also because they are useful in many areas of molecular biology.

Also in protein design they are considered of importance, as different structures related with different repeated structural motifs have generated significant interest with respect to protein engineering and synthetic protein design (Javadi and Itzhaki 2013) and there are various articles where we can find information about re-engineering of TRP binding specificities, with particular attention paid to protein folding kinetics and protein stability.

Moreover, Tompa recognized that many important functions are also linked to proteins that lack a folded structure, and that these functions are not only linked to a well-defined 3D structure protein conformation (Tompa 2002). Furthermore, usually the concept of ordered and disordered proteins is linked to the concept of absence or presence of repeat segments at sequence level, we could say that as a consequence, TRPs are also important to understand protein functions (Tompa et al. 2009).

2.2. Protein repetitive motifs, domain repeats and their evolution

A protein motif is a supersecondary structure containing multiple secondary structure elements in a stable arrangement, not necessarily having a similar function. An example of this is the omega loop, that is a protein motif that resembles the greek letter omega. This makes the motif extremely common, but nothing can be said about its significance in the protein function. On the other hand, a protein domain is a stable structure with a specific defined function in protein and it can exist independently of it, and could maintain its structure even if it is separated from the entire protein. For example :SH2 (src homology 2) domains are found in signalling pathways in JAK-STAT that are responsible for controlling transcription of certain genes. The function of this domain is to bind proteins containing this domain to specific sites in membrane protein (Marco Milán et al., n.d.).

Many large proteins have evolved by internal duplication and many internal sequence repeats correspond to functional and structural units. Evolution

modifies and recombines existing building blocks instead of inventing everything from scratch (Heger and Holm 2000). Protein domain repeats (PDR) are evolutionarily related units that occur in a protein. Many proteins are composed of functional units of common origin. PDR are stretches of domains from the same family, one next to each other in a protein. Structurally, these kinds of domains are diverse and may form modular structures on their own or form larger filaments where each repeat is dependent on the other for functioning. Their sequence is malleable with regard to the repeat unit and the number of repeats, therefore provides flexibility binding to many partners.

Motif and domain are two different concepts that could be related, as is the case of zinc finger domains, an important group of regulatory proteins. Due to their evolution, its sequences contain a variable number of different short sequences with specific symbols at each position, known as zinc fingers (Figure 15).

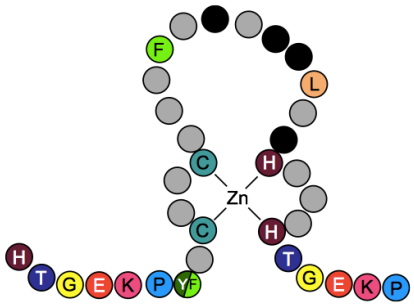


Figure 15 Representation of a structure of a zinc finger

In general, internal duplications in proteins may be grouped in three categories (Katti et al. 2000):

a) Each of the duplicated domains is a structurally and functionally independent unit and possible originates from entire exon duplication (zinc finger).
b) Repeats of 20-40 residues each of which form structurally distinct units (leucine rich) but may not function in isolation.
c) Tandem repeats of single amino acids or short oligopeptides, unlikely to form independent structurally units.

Proteins evolve through mutation and by domain rearrangement. In the case of domain rearrange, mutation tolerance is pretty high because domains perform

modular functions. Repeat proteins vary a lot in the number of repeat units inside the protein. They are different from other proteins by expanding through internal duplication rather than domain shuffling.

a) Optimal RMSD for all unit structures

RMSD matrix (based on multiple superpostion):

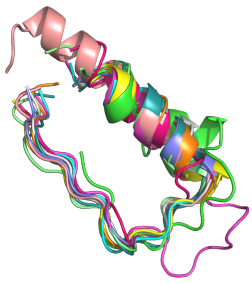
	1	2	3	4	5	6	7	8	9	10	11
1	---	3.4	3.4	3.1	3.6	3.3	4.1	3.7	3.9	3.9	3.1
2	3.4	---	1.0	1.1	1.3	1.2	2.0	1.1	1.6	1.5	2.4
3	3.4	1.0	---	1.4	1.0	1.6	1.7	1.5	1.4	1.5	2.5
4	3.1	1.1	1.4	---	1.6	1.2	2.4	1.3	1.9	1.8	2.3
5	3.6	1.3	1.0	1.6	---	1.3	1.8	1.3	1.0	1.1	2.3
6	3.3	1.2	1.6	1.2	1.3	---	2.3	0.9	1.7	1.6	1.9
7	4.1	2.0	1.7	2.4	1.8	2.3	---	2.3	2.0	2.0	3.0
8	3.7	1.1	1.5	1.3	1.3	0.9	2.3	---	1.7	1.7	2.2
9	3.9	1.6	1.4	1.9	1.0	1.7	2.0	1.7	---	0.9	2.5
10	3.9	1.5	1.5	1.8	1.1	1.6	2.0	1.7	0.9	---	2.5
11	3.1	2.4	2.5	2.3	2.3	1.9	3.0	2.2	2.5	2.5	---

b) Optimal RMSD for unit structures without insertions

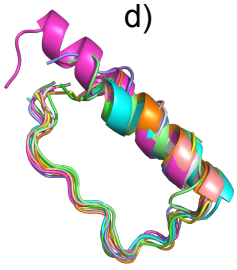
RMSD matrix (based on multiple superpostion):

	1	2	3	4	5	6	7	8	9
1	---	1.3	1.5	1.2	1.7	1.1	1.8	1.5	2.3
2	1.3	---	1.6	1.4	2.3	1.4	1.9	1.9	2.2
3	1.5	1.6	---	1.3	1.7	1.4	1.0	1.2	2.2
4	1.2	1.4	1.3	---	1.9	0.9	1.8	1.5	1.9
5	1.7	2.3	1.7	1.9	---	1.8	2.1	2.0	2.7
6	1.1	1.4	1.4	0.9	1.8	---	1.8	1.6	2.1
7	1.8	1.9	1.0	1.8	2.1	1.8	---	1.1	2.4
8	1.5	1.9	1.2	1.5	2.0	1.6	1.1	---	2.4
9	2.3	2.2	2.2	1.9	2.7	2.1	2.4	2.4	---

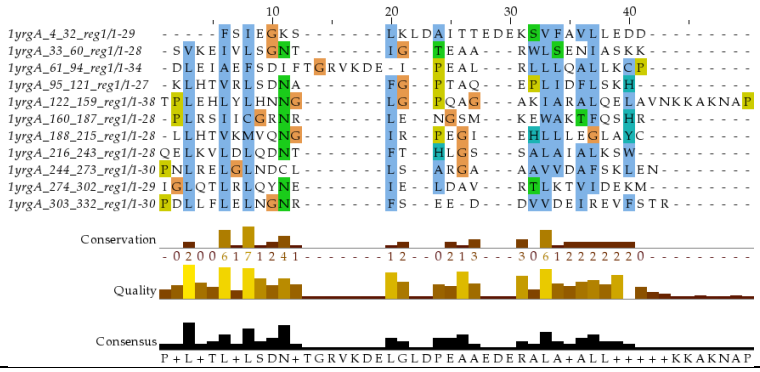
c)



d)



e)



f)

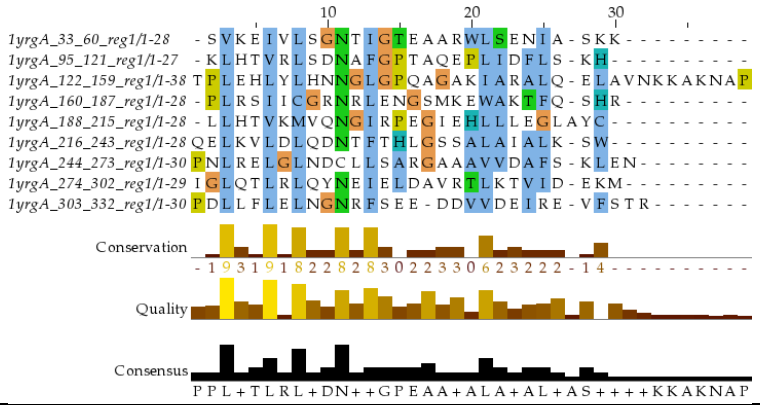


Figure 16 Tandem repeat protein

1YRG chain A a) RMSD matrix for all repeat protein unit, less repeat purity b) RMSD matrix for repeat protein unit without insertions, more repeat purity c) All protein units structurally aligned d) units without insertions structurally aligned e) corresponding sequence alignment of all the repeat units inside the chain f) Sequence alignment of all the units without insertions.

It is possible that repeat proteins expand pretty fast until a physical/structural limit has been reached and as a consequence diverge rapidly since repeat domains tend to only have weak the sequence similarity(Andrade et al. 2001).
Another interesting concept is repeat purity; it is defined as the average pairwise sequence identity between all individual repeat units inside one protein. By the use of artificial proteins and zone domain repeats it has been demonstrated that above 40% sequence identity, protein aggregation and misfolding become very likely (Wright et al. 2005). In Figure 16, protein 1yrgA is shown considering all the repeat units a) c) e) while in b) d) f) show only the units that will allow us to have a high repeat purity (no insertions). To obtain these images we use Mustang (Konagurthu et al. 2006), a) and b) show the Optimal RMSD values using as input all (11) repeat units and the units without insertions. Then in c) and d) we can observe how the units are structurally align ed and comparing them we can observe and identify where the insertions are. Finally e) and f) present the corresponding sequence alignment and in the case of f) we are able to observe conservation of residues that were not observed in e) as a result of the low repeat purity.

2.3. Structural classification of repeats

Appraisal of known protein structures and their classification uncovers a straightforward relationship between their architecture and the length of the repetitive units. This relationship and the repetitive character of structural folds suggest rules for better prediction of 3D structures of such proteins (Andrey V Kajava 2012).

More than fifteen years ago, a classification of 3D structures based on the repeat length was suggested by A. Kajava (A. V. Kajava 2001). Ten years later, the appearance of new 3D structures allowed a refinement (Andrey V Kajava 2012) shown in Figure 17.
In 2012 structural classification of five different classes and their subclasses were identified:

- Class 1: Crystalline aggregates of unlimited size

Includes proteins and peptides with 1 or 2 residue-long repeats that forms different types of crystallites which are harmful to living organisms. The regions of proteomes with such repeats have propensity to be unfolded and are mostly hydrophilic (Julien Jorda et al. 2010). The structures of these protein regions are absent in the PDB. All the knowledge about them is from previous studies. Huntington's disease, a human neurodegenerative disorder is caused by expansion of polyglutamine, and molecular packing of polyglutamine is related to this class, which make it an interesting subject of study.

- Class 2: Fibrous structures stabilized by interchain interactions
  In this class we find collagen and α helical coiled coils, two major fibrous structures. Collagens have a tripeptide repeat Gly-X-Y, where X and Y could be any residue but are usually proline or hydroxyproline.

- Class 3: Elongated structures where repetitive units that require one another to maintain structure.
  In this class we can find solenoids and non-solenoids structures.
  - Solenoids: Based on solenoidal windings of the polypeptide chain. They tend to have elongated structures and are predominant in this class.
    - β-Solenoids (III.1)
    - α/β-Solenoids (III.2)
    - α-Solenoids (III.3)

  - Non-solenoids: In this subclass we find more complicated folds than solenoidal fold.
    - Single layer antiparallel β (III.4)
    - Trimer of β-spirals (III.5): with long central β strands that hold the trimer together through interchain hydrogen bonds, and interactions of apolar side chains and short peripheral β strands stabilize the structure.

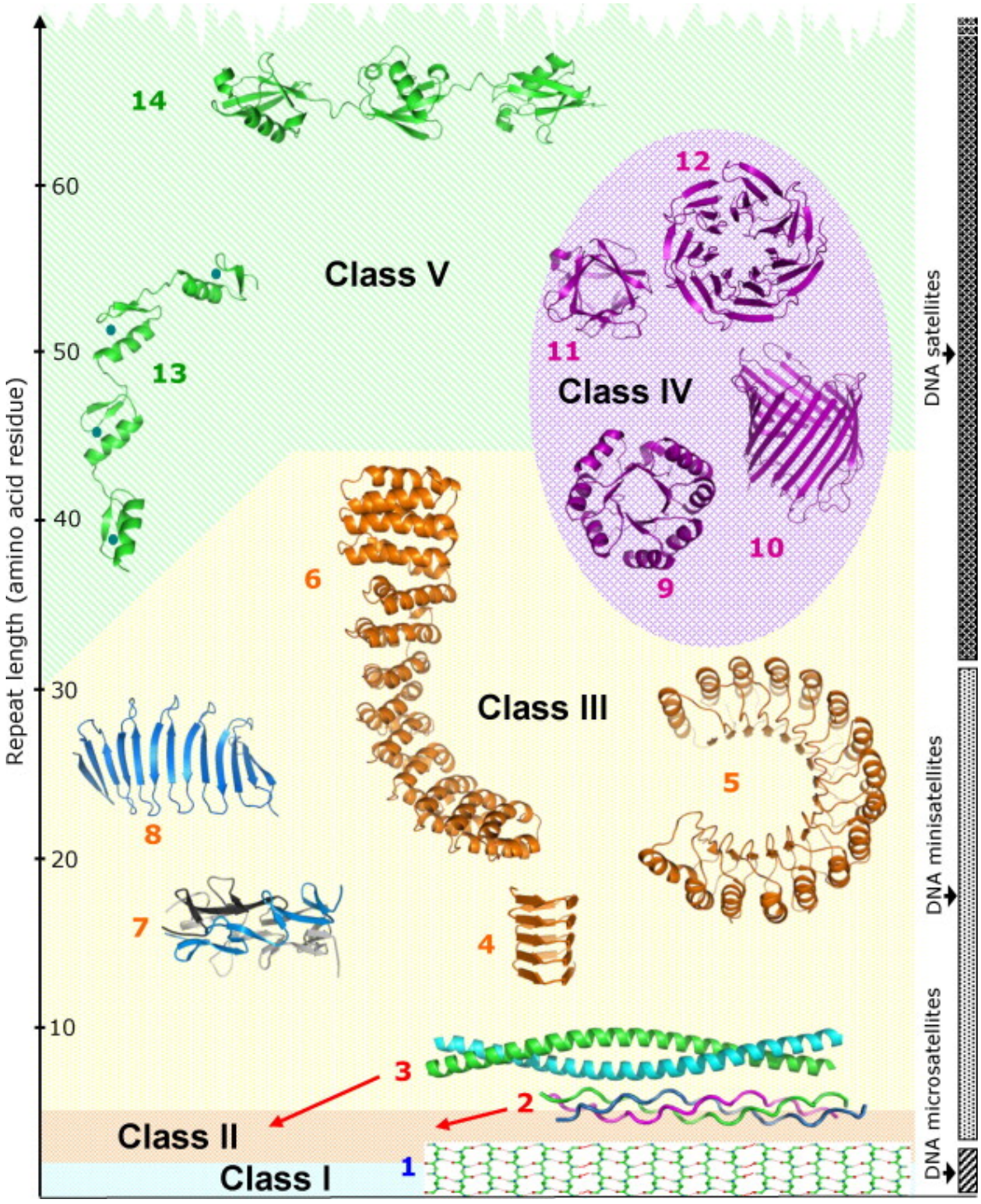


Figure 17 Structural Classification

- Class 4: Closed structures where repetitive units need one another to have structure.

  In contrast with previous classes, that are able to have an unlimited number of repeats and do not have restrictions to axial growth these called closed structures have a fixed number to make them “closed”.

- Tim Barrel (IV. 1)
- β Barrel (IV. 2)
- β trefoil (IV. 3)
- β Propeller (IV.4)
- α/β prism (IV.5)
- α barrel (IV.6)

- Class 5: Beads on a string structures which repeats are large enough to fold independently.

  In this class we can find repetitive units, which are large enough to fold independently into stable domains. They usually have from 50 – 60 residues.
  - α bead (V.1)
  - α/β beads (V.2)
  - β beads (V.3)

2.4. Identification of repetitive elements

Considering classes III and IV from the structural classification, a "repeat" protein is a protein containing a repeat domain, composed of repeating homologous structural units (repeats), which tightly stack together forming a joined hydrophobic core. The stability of the domain is ensured by mutual stabilization of the repeats. A repeat is one of several repeated homologous building blocks of a repeat domain. It has a well-defined topology when present in a repeat domain, but is usually unfolded on its own (Forrer et al. 2004).

Many repeats appear to possess high amino acid substitution rates and thus recognition of repeat homologues is highly problematic (Andrade et al. 2000). Moreover, detection of this kind of proteins is another challenging task. In literature we can find algorithms based on sequence and based on structure. Even in the presence of high divergence among subsequences corresponding to TRP units, DNA coding sequence and amino acid sequences are usually

preserved. Since 1990, different approaches have been developed and most recently there is a tendency to integrate basic sequence data with evolutionary or biochemical annotation. M. Andrade presented a detection algorithm which uses a homology based method to identify statistical significance in protein repeats (Andrade et al. 2000). Other methods are based on detecting suboptimal alignments in self-alignment matrix generated by Smith-Waterman or a similar method(Heger and Holm 2000)(Szklarczyk and Heringa 2004)(George and Heringa 2000).

There are other methods that use a seed expansion (Newman and Cooper 2007) approach while others use a clustering approach based on k-means(J. Jorda and Kajava 2009). In literature we can also find some approaches that are based on building, matching Hidden Markov Models for the repeating substring(Soding, Remmert, and Biegert 2006) and others based on neural networks that aim to detect a specific repetitive structure. The sequence-based algorithms are many, each has a different approach and to our knowledge there is no comparative study. It is not surprising that sequence-based methods fail to infer true structural repetitions since the same structural motif can be encoded by sequences that appear completely unrelated, which is the case in several repeat-protein families. Thus sequence approach is not usually the best one, but when there is no structure it should be enough.

Its already known that function features are more linked to structure of the protein than to primary sequence thus available structural data should be used to detect repetitive motifs or units. Of course there are some approaches that use both, sequence and structure signals as in (Murray, Gorse, and Thornton 2002). And other approaches that integrate structural information with other methods as Fourier transform (Murray, Taylor, and Thornton 2004) or dynamic programming (Sabarinathan, Basu, and Sekar 2010). There are also various other tools that are focus in a specific class of repeat structures, like Raphael (Walsh et al. 2012) and (Hrabe and Godzik 2014) created for detection of solenoids.

3. Methods

3.1. Predicting repeat units with ReUPred

Numerous studies in biological processes have been made over the last 10 years and they have demonstrated the importance of tandem repeat proteins in these processes as mentioned in (Di Domenico et al. 2014). In 2014, Di Domenico et al. published RepeatsDB: a database of tandem repeats protein structures. The database was one of the first efforts to systematically classify and annotate structural protein repeats in a consistent way. It provides a detailed structural characterization of the repetitive elements (Figure 18), which includes the PFAM domains, the repeat units, and the corresponding fragments in the sequence but only for 3% of the entries (Table 2) (Di Domenico et al. 2014a). In this database we were able to find more details on structural classification made by Kajava including the five classes and the subclasses (Andrey V Kajava 2012) (Figure 19). Kajava based this classification on the repeat unit length, but only a small number of protein repeat units in each subclass were manually identify as mentioned before. In Table 3, we show statistics for 2014 RepeatsDB, not only entries with detailed information (Figure 20) but also entries predicted as repeat by Raphael. Raphael is a method for the detection of solenoids in protein structures. Its reliability solves three problems of increasing difficulty: recognition of solenoids domains, determination of their periodicity and assignment of insertions (Walsh et al. 2012).

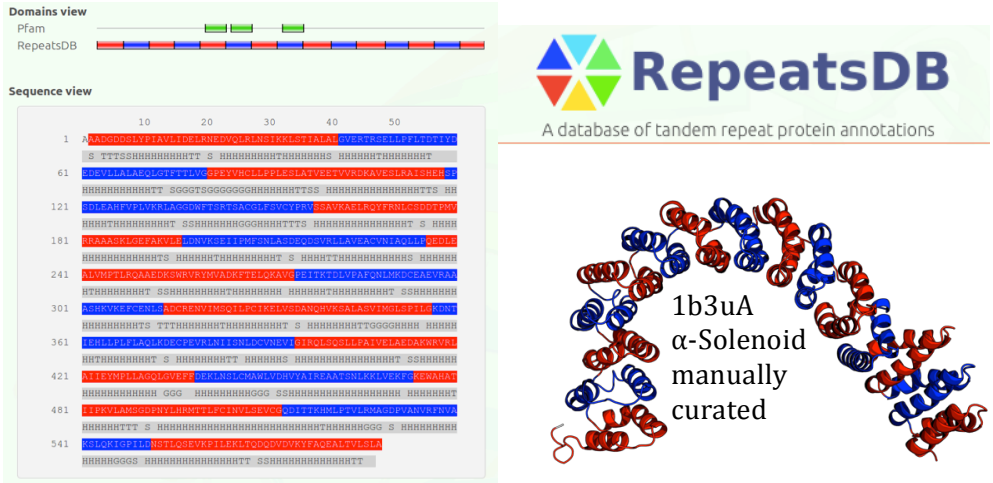


Figure 18 RepeatsDB 2014 entry

The method recognized repeat proteins but without detailed information of the units. This amount of detailed information was not enough for any reliable large-scale analysis. To increase RepeatsDB annotation and to solve the problem of modeling repeat proteins we adopted an innovative approach using detailed information of Solenoids. Initially using all fragment units we created a Structural Repeat Unit Library (SRUL). To create it, we only considered units with at least 10 residues. However, we retained all the rest of the units, even the ones with insertions and finally to avoid redundancy we considered 90% of sequence identity between the remaining units.

Table 2 Statistics by class of RepeatsDB 2014

	Detailed	Classified (manually)	Classified (By similarity)	Predicted
I - Crystalline aggregates	0	0	0	0
II - Fibrous structures	23	41	69	0
III - Elongated structure	119	397	692	0
IV - Closed structure	149	300	890	0
V - Beads on string	36	16	76	0
UA - Unassigned	0	0	0	7948
Total	327	754	1727	7948
Total (%)	3%	7%	16%	74%

Table 3 Statistics by subclass of RepeatsDB 2014

Subclass	Detailed	Classified (manually)	Classified (By similarity)
III.1 - β-solenoid	41	108	21
III.2 - α/β-solenoid	19	43	27
III.3 - α-solenoid	48	244	631
III.4 - trimer of β spirals	7	0	13
III.5 - single layer β	4	3	0
IV.1 - TIM-barrel	84	117	626
IV.2 - β-barrel	8	1	8
IV.3 - β-trefoil	15	0	29
IV.4 - β-propeller	38	168	227
IV.5 - α/β-prism	0	14	0
IV.6 - α-barrel	5	0	0
V.1 - α-beads	2	1	0
V.2 - β-beads	29	12	71
V.3 - α/β-beads	3	3	1

After having a clean version of the SRUL we created ReUPred (L. Hirsh et al. 2016). Repeat Protein Unit Predictor is a novel method for fast automatic

prediction of repeat units and repeat classification and its accuracy depends on the Structure Repeat Unit library (SRUL). The SRUL library contains a set of fragments, one for each of the repeated units identified.

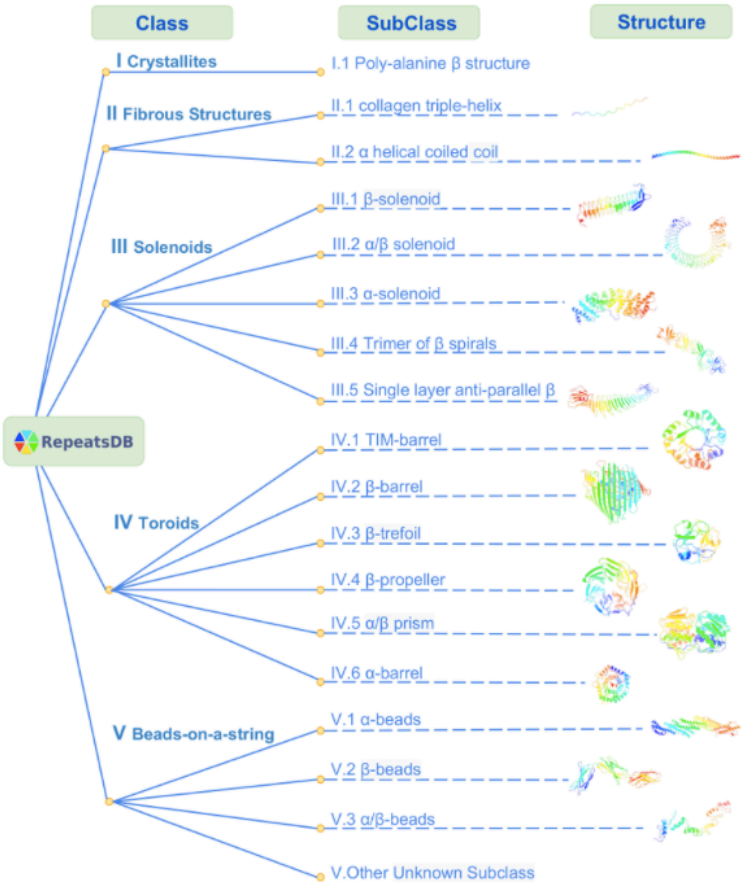


Figure 19 RepeatsDB 2014 Structural classification

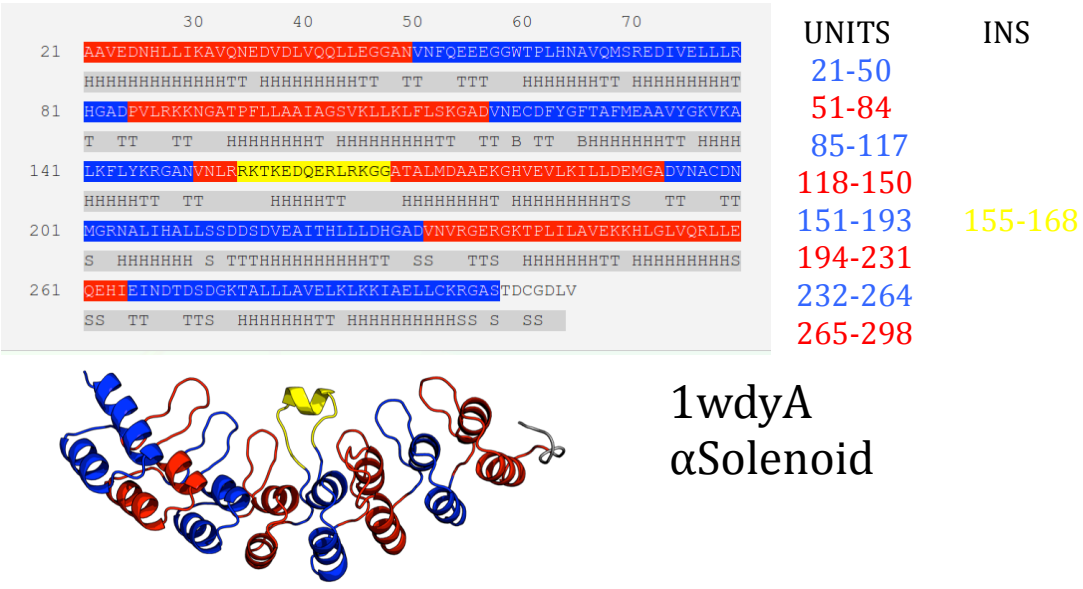


Figure 20 RepeatsDB 2014 detailed entry example

ReUPred uses an iterative structural search against the SRUL to find a master repetitive unit. This called master unit is the first repeat unit found in the target. Therefore, it is the result of structural alignment of SRUL units against the target protein. Then it uses the master unit on the target protein to find all the rest of the repeat units based on best structural alignment. We use 2v70 protein (Third LRR domain of human SLIT2), chain A as a practical example of ReUPred method.

In Figure 21, we can observe four examples of alignments between target (2v70A) and four different units of SRUL, as shown in all images, master unit is option A with a TMscore of 0.69, and the used template is 1oznA unit 7 (Figure 22). We choose this unit template because it covers the thresholds (Table 4) and it has the higher TM-score value. The TM-score value is a value between 0 and 1, higher the value, better the structural alignment (Figure 23).

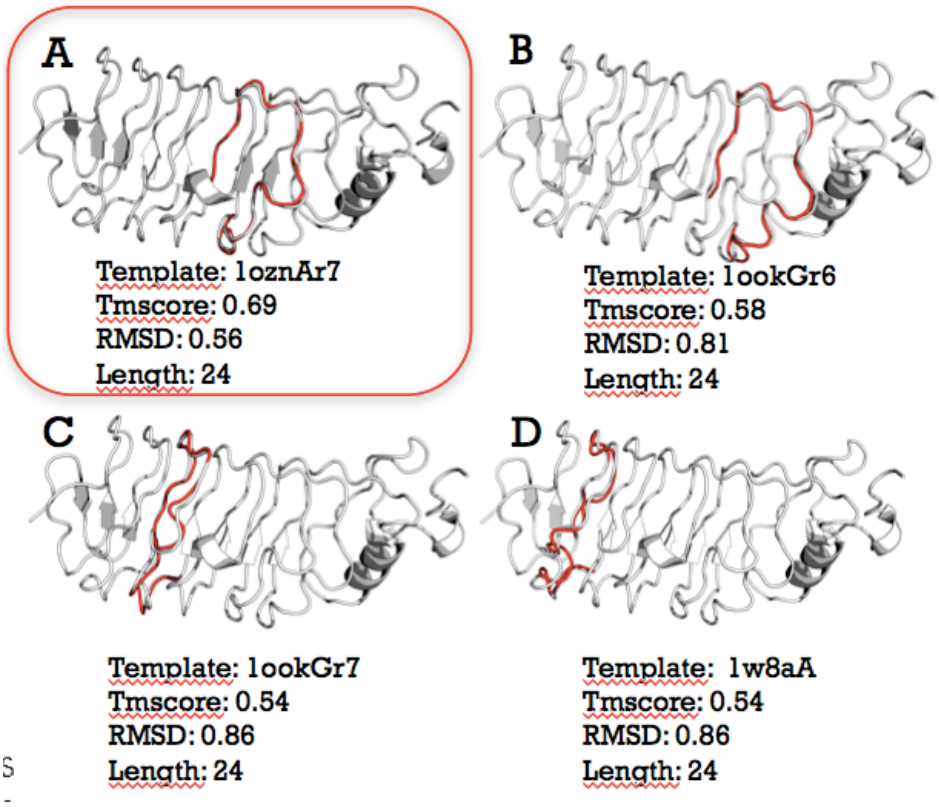


Figure 21 ReUpred selection of master unit
Different alignments of target protein against SRUL units in where 1oznA_unit7 is the best template unit.

a)

b)

Figure 22 Protein 2v70 chain A
a) Structure of ReUpred target protein b) Sequence fragment resulting of structure alignment of the target protein versus the template from the SRUL library

Table 4 Thresholds for the target protein against SRUL units

Iteration	TM-Score	RMSD (A)	Alignment (residues)	Unit gaps (%)
1	≥ 0.52	≤ 1.6	> 21	< 10
2	≥ 0.47	≤ 1.9	> 17	< 20
3	≥ 0.30	≤ 2.5	> 16	< 50
4	≥ 0.23	≤ 3.0	> 14	< 50

Figure 23 Different results of aligning two unit structures

In Figure 24, we can see the structure of the called master unit (610 - 633 residues of 2v70A), and how extracting this master unit from the protein, two different fragments are created, one from 505 - 609 residues and the other from 634 - 714 residues.

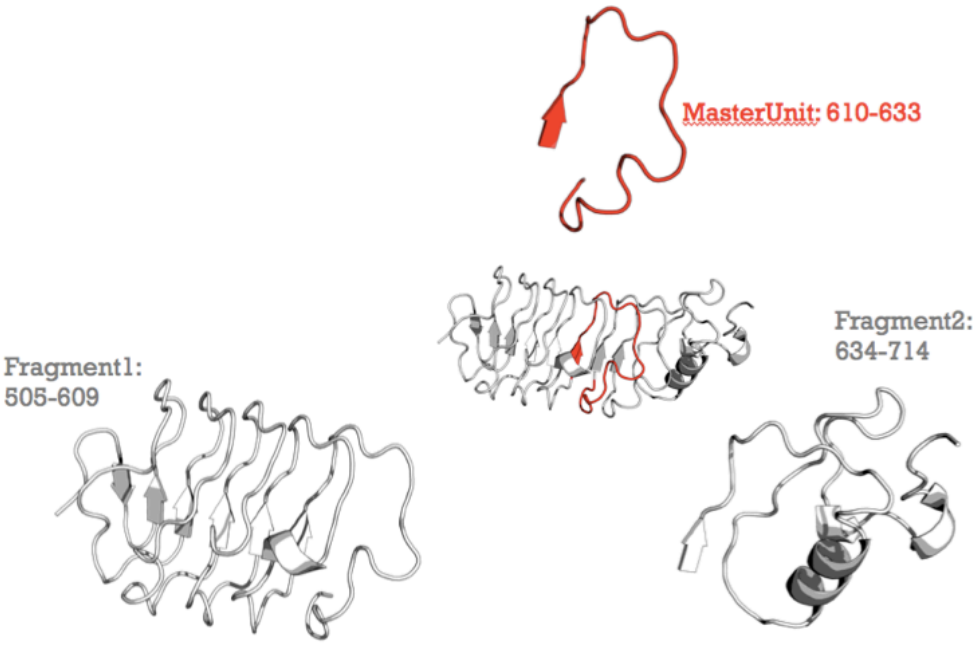


Figure 24 ReUpred master unit and remaining fragments

Then, the so called master unit is aligned with both fragments, and we select the best (the one with a higher TMscore) (Figure 25). In this case, selection is based in the RMSD and TMscore value and the corresponding threshold values(Table 5), as both of alignments have almost the same value for TMscore we choose the one with a lower RMSD value.

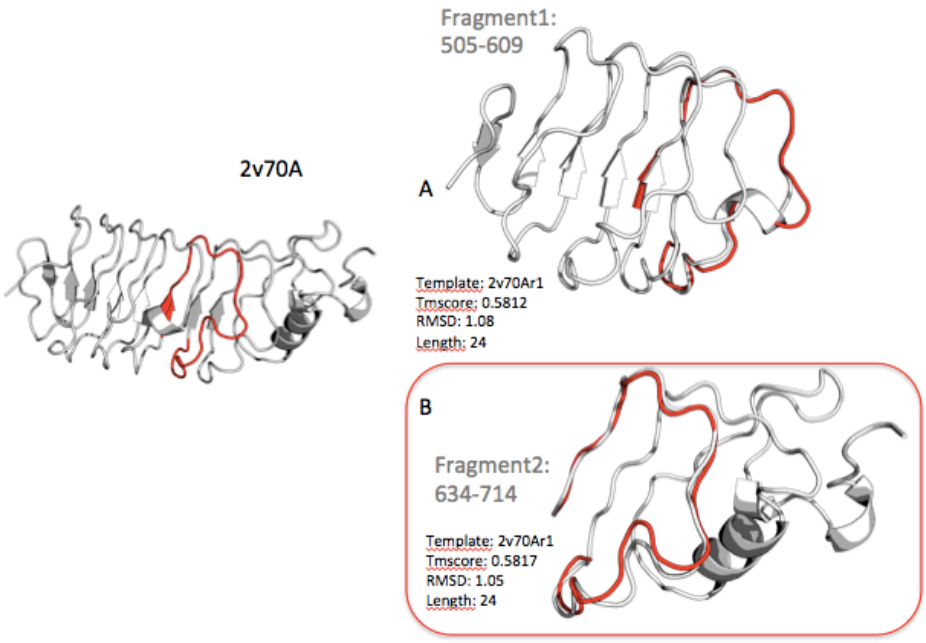


Figure 25 ReUpred Selection of master unit

Then we repeat the previous process creating new fragments and aligning the fragments with previously created repeat units of the target protein, until there are no possible segments to evaluate or until alignments do not reach the thresholds. In Figure 26 we can see how units colored in blue and red are the predicted units and how there are regions in grey that do not reach the threshold so they are not considered as part of the repeat region.

Table 5 Thresholds for predicted units inside a protein

Iteration	TM-Score	RMSD (A)	Alignment (residues)	Unit gaps (%)	Length ratio (%)
1	≥ 0.35	≤ 1.8	≤ 1.20	< 40	≥ 70
2	≥ 0.30	≤ 2.0	≤ 1.15	< 40	≥ 70
3	≥ 0.30	≤ 2.5	≤ 1.15	< 40	≥ 70
4	≥ 0.30	≤ 3.0	≤ 1.10	< 50	≥ 70

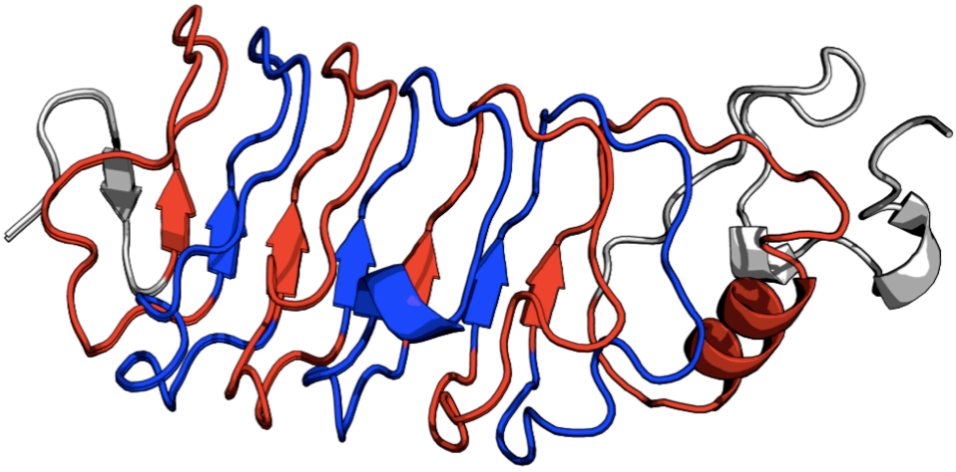


Figure 26 ReUpred repeat units

For assigning the prediction subclass, ReUPred does the classification using the class of template unit selected from SRUL library. For the shown example we use 1oznA part of the α/β-solenoid so 2v70A would be classified by ReUPred as an α/β-solenoid.

To evaluate the resulting prediction we had to redefine some concepts applied to the repeat unit idea. So, as shown in Figure 27, and based on an evaluation (eval) of a prediction (Pred) against a manually curated reference (Ref) we obtained true positives (TP), false positives (FP), true negatives (TN) and false

negatives (FN) are calculated. For classification, given a solenoid subclass, TP is the number of proteins with correct assignment, FN are proteins assigned the wrong class and FP are class assignments to wrong targets. TN is always zero since the test set contains only classified proteins. For all evaluations, the measures recall [or sensitivity; TP/(TP + FN)], precision [TP/(TP + FP)] and accuracy [(TP + TN)/(TP + FP + TN + FN)] are used.

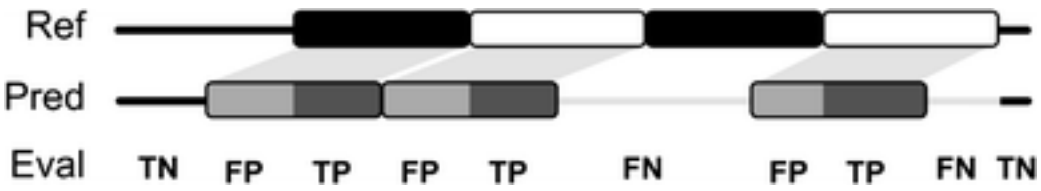


Figure 27 Evaluation for repeat unit predictor

3.2. Large scale annotation

In 2016 ReUPred (L. Hirsh et al. 2016) was created, originally for only solenoids structures but recently updated the method, SRUL library and thus optimizing the algorithm. So, as in the case of solenoids version of the predictor, we started the process creating the SRUL library, but this time using all detailed information in classes III, IV and V of RepeatsDB 2014 (Table 6).

Table 6 Statistics by class of RepeatsDB 2014, detailed information

	Detailed	Classified (manually)	Classified (By similarity)	Predicted
I - Crystalline aggregates	0	0	0	0
II - Fibrous structures	23	41	69	0
III - Elongated structure	119	397	692	0
IV - Closed structure	149	300	890	0
V - Beads on string	36	16	76	0
UA - Unassigned	0	0	0	7948
Total	327	754	1727	7948
Total (%)	3%	7%	16%	74%

Using 304 detailed entries, we manually curated the data and created a new version of SRUL. This curating process included the redefinition of insertions (Figure 28), reclassification of proteins (Figure 29) and even the definition of new regions inside a protein (Figure 30).

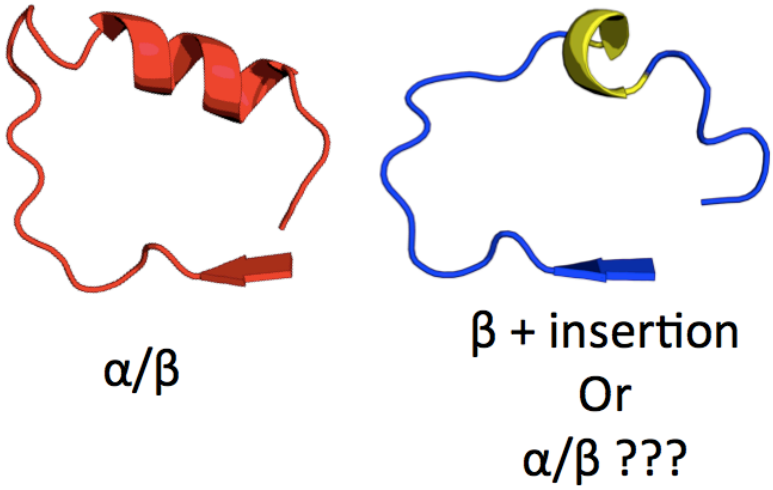


Figure 28 unit classification possible error

The resulting and original entry values for SRUL are shown in Table 7. Then using this version of the library we updated the ReUPred algorithm (Figure 31). To our surprise, looking at some of the new predictions, we observed that it wrongly classified some repeat structures. These repeat structures were wrongly classified because there was no similar subclass defined in Kajava's stucture definition(Bostjan Kobe and Kajava 2000). But considering the logic of the algorithm ReUPred correctly identifies the protein as a repeat.

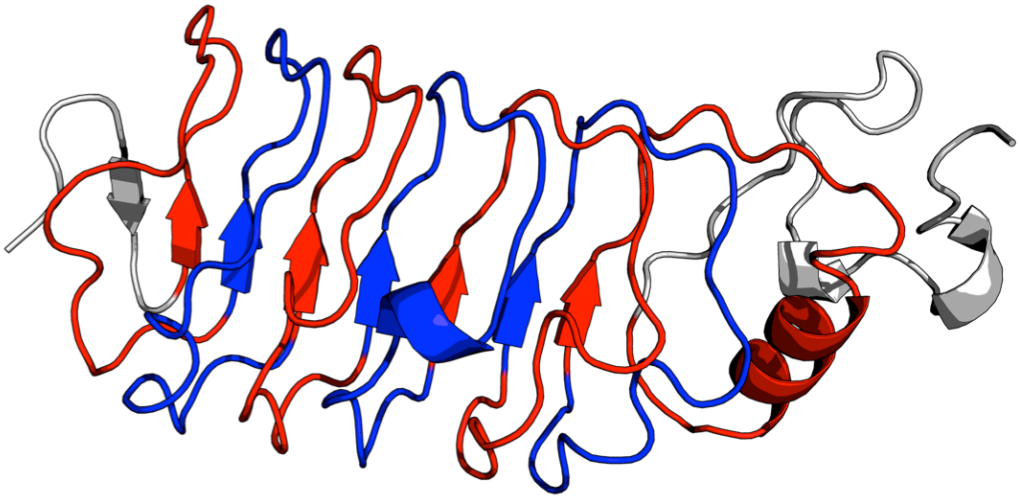


Figure 29 Protein reclassification possible error

Table 7 Statistics modifications of the new SRUL

	Number of chains	
Subclass	**SRUL**	**SRUL rev.**
III.1 - β-solenoid	41	39
III.2 - α/β solenoid	19	21
III.3 - α-solenoid	46	47
III.4 - trimer of β spirals	7	7
III.5 - single layer β	4	3
IV.1 - TIM-barrel	84	83
IV.2 - β-barrel	8	6
IV.3 - β-trefoil	15	13
IV.4 - β-propeller	38	37
IV.5 - α/β prism	0	2
IV.6 - α-barrel	5	5
V.1 - α-beads	2	2
V.2 - β-beads	29	25
V.3 - α/β-beads	3	4
V.Other	3	0
Total	**304**	**294**

Modification	Number of chains
Switch subclass	9
Remove	15
Add	4
Split	1
Join	2

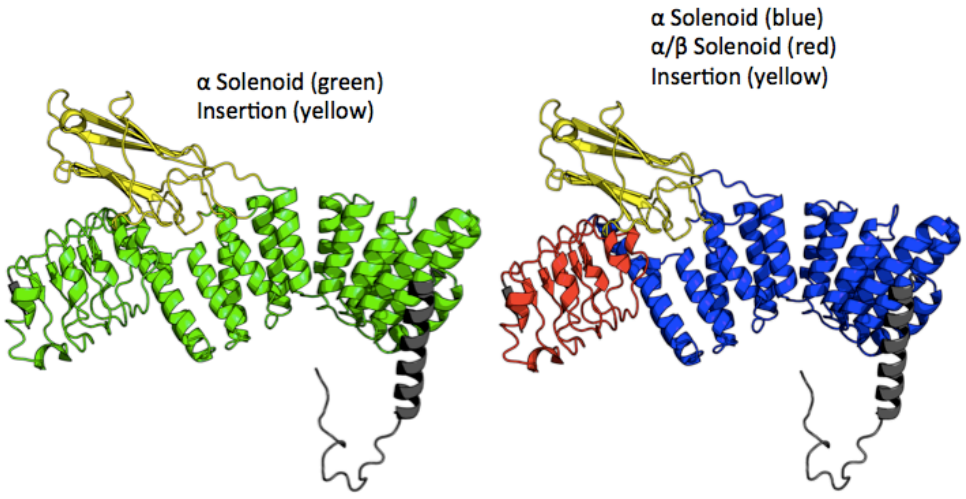


Figure 30 Region classification possible error

Consequently, to create new structural subclasses we used the process shown in Figure 32. In it we gathered a set of proteins using Raphael(Walsh et al. 2012) predictions and then we use them as inputs for ReUPred, analyze the predictions and create new structural subclasses when needed and add new repeat units to SRUL. Finally we run ReUPred against the PDBdata bank to create the data inside the latest version of RepeatsDB (Paladin et al. 2016).

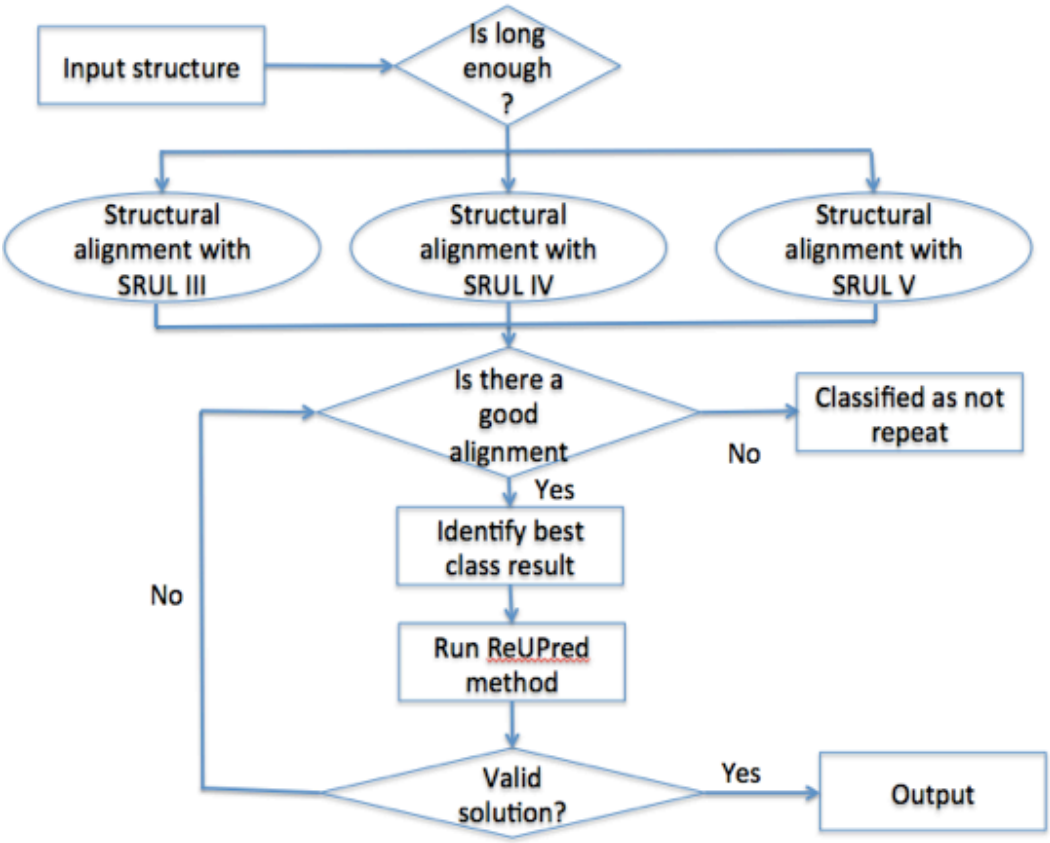


Figure 31 ReUpred 2.0 algorithm

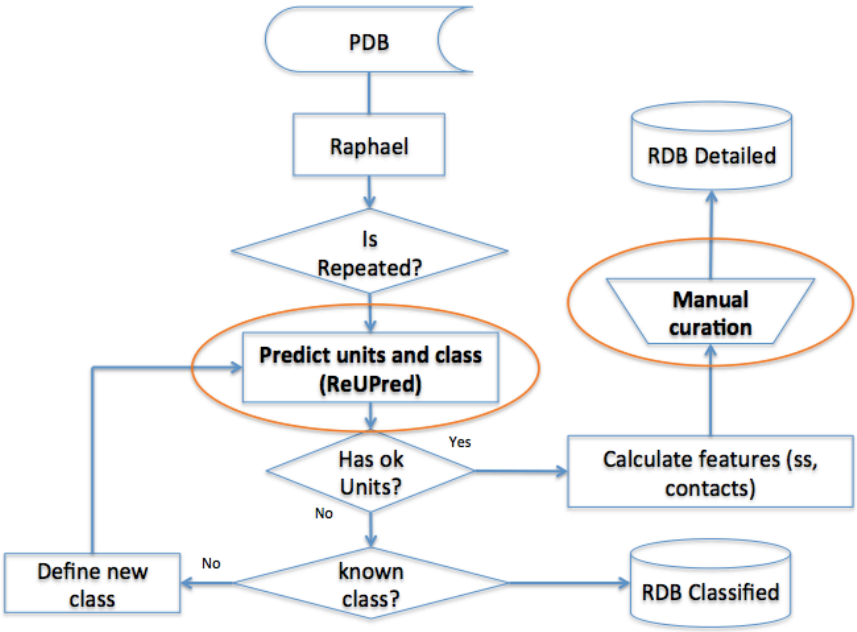


Figure 32 Data creation process

This RepeatsDB updated version presents an improved annotation, classification, search and visualization. In this database, on the contrary to the previous version, all entries have unit information and it is mapped with other repositories, allowing scientifics to have everything they may need.

3.3. Solenoid ensembles

In 2016 we created and published RepeatsDB 2.0 (Paladin et al. 2016), in this database we were able to find 5498 entries corresponding to 970 UniProtKB sequences. From this 5498 entries 3307 were manually curated and the rest were ReUPred predictions. To further improve RepeatsDB's quality we decided to provide a finer classification at subclass level. We focused on α-solenoids that represent the most abundant fold in repeat proteins. α-solenoids are flexible protein structural domains formed by ensembles of α-helical repeats. Ensembles as HEAT, Armadillo, TPR among others inside this type of solenoids do not have the same structure or sequence between repeat units. Moreover, this kind of protein adopts a variety of elongated curved structures and functions. We initially hypothesized that inside ensembles of α-solenoids proteins, i.e. at sub class level, repeat units would share a similar structure that should let us characterize a specific ensemble. To verify this hypothesis we did an analysis of repeat units inside this subclass. Taking into consideration the remaining 740 repeat units after curation process, we created a matrix of all versus all repeat units, in which for each of them we calculate structural alignment values (TMscore) between all the rest of the units. Then we created a network using each of the units as a node and each of the corresponding structural alignment value as an edge. The process followed for this curation is shown in Figure 33, first we collected all the 1329 α-solenoid entries from RepeatsDB, and randomly select and evaluate some of them.

As shown in Figure 33, we first create the repeat unit, evaluate the entry using structure alignment of the repeat units and their secondary structure, we also identify missing segments in the structure and visually verify if the units were rightly defined. Then we create the dataset with entries that were considered as valid. Some of them share same UniProtKB sequences which could mean a

reduction of time in the verification process but similar UniProtKB sequence is not necessarily a similar secondary structure as shown in Figure 34.

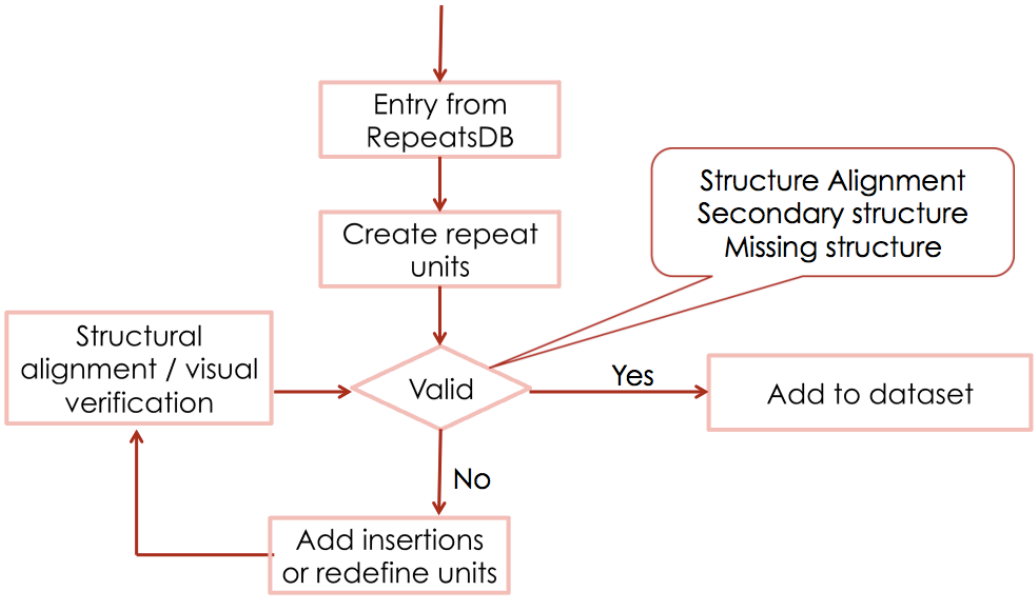


Figure 33 Protein curation process

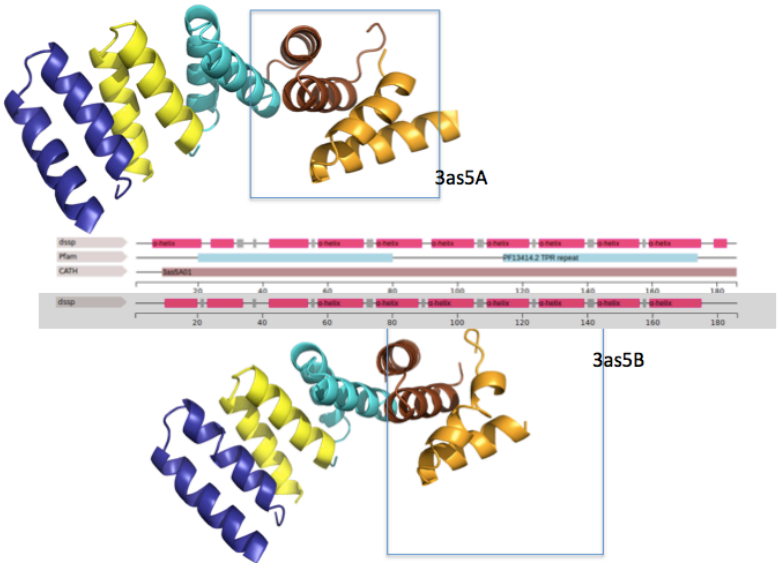


Figure 34 Protein chains with same UniProtKB sequence
Protein 3as5 chain A and chain B, same sequence different UniProtKB

We wanted a unified unit structure definition for each type of α-solenoids proteins, and have the same time a definition that allows us to differentiate one ensemble from the other. Which led us to a possible redefinition of some of the units and/or inclusion of insertions as shown in Figure 36.

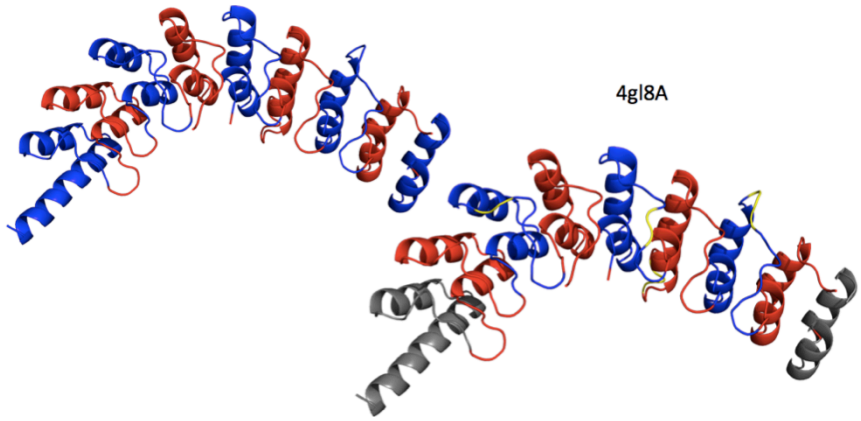


Figure 35 Filtered border units and units with insertions

For this redefinition, we always try to reduce the length of loops inside the unit, maintaining a standard of having the longer α helixes in the N terminal and C terminal, having two α helix bonded by loops, or three α helix where the smallest helix would be in the middle. The reason for this decision was mostly because we were not able to decide if the small α helix should be in the N terminal or in the C terminal (Figure 36).

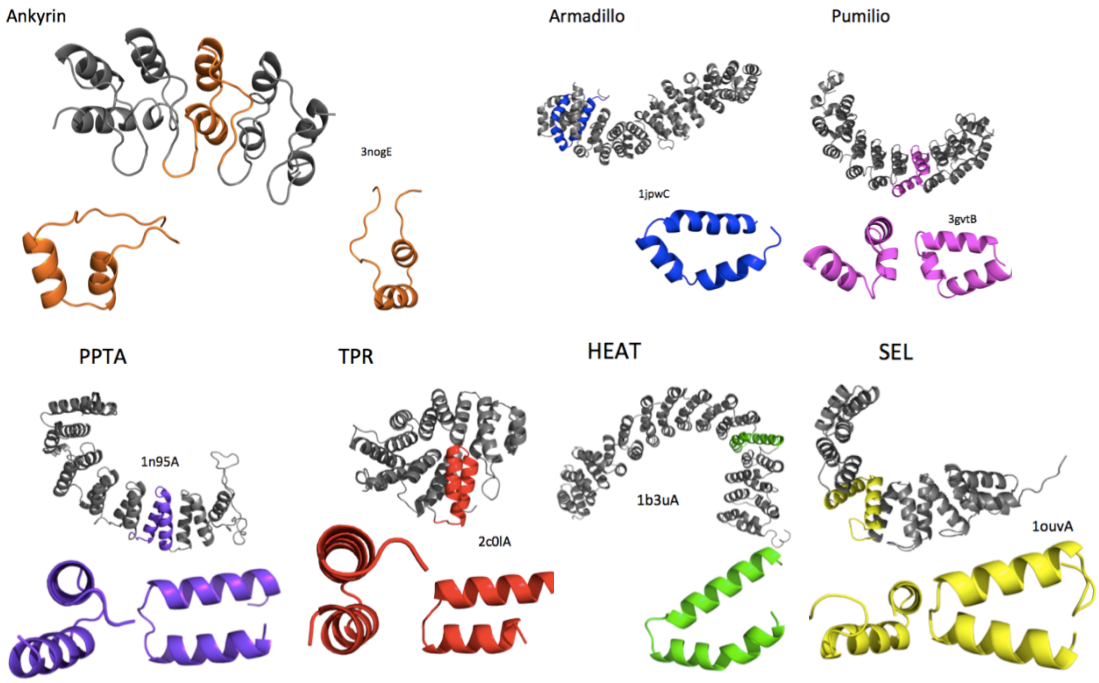


Figure 36 Protein unit structure definition

Finally, after this curation process we got a dataset of 620 entries, which means 4699 repeat units. Using Victor library (L. Hirsh et al. 2015) we acquired all sequences that were the input for CD-Hit at 100% to reduce the repeat units

redundancy, and obtained 1193 units without redundancy in which we based our analysis.

We aligned all the units structurally and considered a threshold of 0.7 of TM-score, create the network shown in Figure 37.

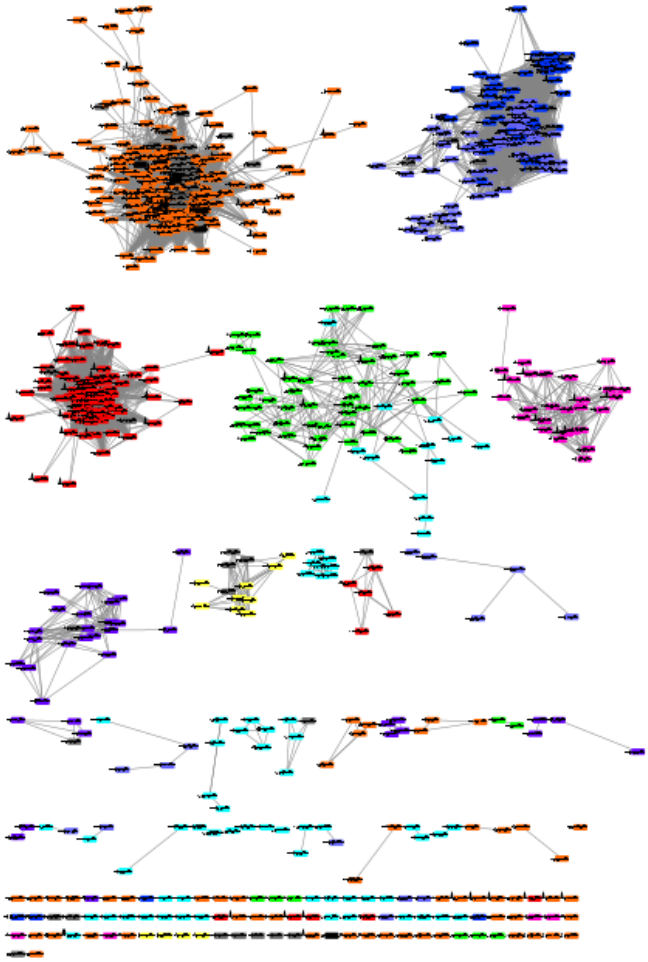


Figure 37 Network created based on 1193 units matrix

3.4. Manipulating proteins structures with VIctual Construction TOol for pRoteins (Victor)

Protein sequence and structure representation and manipulation require software specially made for this end. There are many tools developed but most of them are not open source or they use a lot of computational resources. Instead, we propose an open source library called Victor (L. Hirsh et al. 2015), made in C++ programming language. This library is able to manipulate protein structures with minimal computing time.

Fifteen years ago S. C. E. Tossato created the first version of the library, and for many years after that, students of different levels have been creating and updating different applications and publishing them, demonstrating the effectivity of the never published complete library. But because of the different coding style inside the library a standarization was needed together with the validation of new formats for different possible input files as PDB or Fasta files.
This is the reason why we made this reengineering, which includes not only the standardization of code but also the creation of a unit test for all main methods and the creation of extensive online material; which includes the explanation of how classes work and also instructions for installation and how to use different applications.
The first step for standardization of code was to identify the classes and unify the style of classes' definitions, which include names of attributes, methods, comments and examples. This library contains more than 60000 lines of code and still expanding. The Victor2.0 library (Virtual Construction Toolkit for Proteins) is composed of four main modules:

- Biopool - BIOPolymer Object Oriented Library. Generates the protein object and provides useful methods to manipulate structure.
- Align - ALIGNment generation and analysis.
- Energy - A library to calculate statistical potentials from protein structures.
- Lobo - LOop Build-up and Optimization. Ab initio prediction of missing loop conformation in protein models.

Initially the reengineering process started from Biopool module based in a structural design pattern. These design patterns are all about Class and Object composition. Structural class-creation patterns use inheritance to compose interfaces. Structural object-patterns define ways to compose objects to obtain new functionality(Gamma 1995). A Protein object is just a container for vectors representing chains. Each vector has 2 elements: Spacer and Ligand Set. Spacer is the container for AminoAcid objects whereas LigandSet is a container for all other molecules and ions, including DNA/RNA chains. Ultimately all molecules, both in Spacer and inLigandSet are collections of Atom objects (Figure 38). The main feature in Biopool is that each AminoAcid object in

Spacer is connected to its neighbours by means of one rotational vector plus one translational vector.

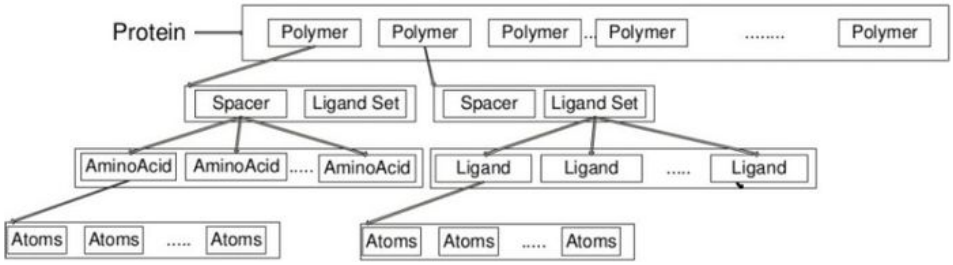


Figure 38 Biopool protein pattern

This implementation makes the modification of protein structure fast and a lot of functions were implemented to, for example modify/perturbate/transformate or residue relative position, in an efficient way. The next reengineered module was Align, as it has the simplest application, even do it has severals options. The necessary data files (e.g. substitution matrices) are provided. The most important feature of the package is the modular object oriented design, which should allow a moderately experienced C++ programmer to rapidly implement and test new features for sequence alignment. Inside this package, different weighting schemes, scoring functions, ways to penalize gaps, and typologies of structural information can be used. The Align library was designed to be modular and easy to expand. There are four basic components which are needed to use alignment methods.

The four main components are:

- AlignmentData - Stores information on sequence (*SequenceData*) and, when needed, secondary structure (*SecSequenceData*).
- ScoringScheme - Stores information on how a single position shall be scored in alignment.
- Align - The alignment algorithm. It requires both *AlignmentData* and *ScoringScheme* objects.
- Blosum - The substitution matrix.

Then one of the most used modules: Energy, has already been used in different publications. Energy functions are used in a variety of roles in protein modelling. An energy function precise enough to always discriminate native protein structures from all possible decoys would not only simplify protein structure prediction problems considerably. It would also increase understanding of the

protein folding process itself. If feasible, one could use quantum mechanical models, being the most detailed representation to calculate the energy of a protein. It can theoretically be done by solving Schrödinger equation. This equation can be solved exactly for the hydrogen atom, but is no longer trivial for three or more particles. In recent years it has become possible to approximately solve Schrödinger equation for systems up to a hundred atoms with Hartree-Fock or self-consistent field approximations. Their main idea is that many-body interactions are reduced to several two-body interactions. The functions of this class are important to all aspects of protein structure prediction, as they give a measure of confidence for optimization. An ideal energy function would also explain the process of protein folding. The most detailed way to calculate energies are quantum mechanical methods. These are, to date, still overly time consuming and impractical. Two alternative classes of functions have been developed: force fields and knowledge-based potentials. Force fields (e.g. AMBER) are empirical models approximating the energy of a protein with bonded and non-bonded interactions, attempting to describe all contributions to total energy. They tend to be very detailed and are prone to yield many erroneous local minima. An alternative is knowledge-based potentials, where “energy” is derived from the probability of a structure being similar to interaction patterns found in the database of known structures. This approach is very popular for fold recognition as it produces a smoother “global” energy surface, allowing detection of a general trend. Abstraction levels for knowledge-based potentials vary greatly and several functional forms have been proposed. The energy functions presented in the package allow optimizing procedures. The main feature is its applicability in the context of protein classes implemented in the package. It should be possible to invoke energy calculation with any structure from all programs. Previously parameters of energy models had to be stored externally to allow their rapid modification. With this considerations in mind, the package Energy was designed to collect classes and programs dealing with energy calculation. The main design decision was to use the “strategy” design pattern from (Gamma 1995). The abstract class Potential was defined to provide a common interface for energy calculation. It contains necessary methods to load energy parameters during initialization of an object.

Computing energy value for objects of Atom and Spacer classes as well as a combination of both is allowed.

Finally but not less important is the Lobo module, current database methods using solely experimentally determined loop fragments do not cover all possible loop conformations, especially for longer fragments. And for sure, it is not feasible to use a combinational search of all possible torsion angle combinations (Figure 39). For an algorithm to be efficient, a compromise has to be found. One improvement in ab initio loop modelling is the use of look-up tables (LUT) to avoid repetitive calculation of loop fragments. LUTs can be generated once and stored; only requiring loading during loop modelling. Using a set of LUTs reduces computational time significantly. The next problem is how to best explore conformational space. Especially for longer loops, it is useful to generate a set of different candidate loops to exclude improbable ones by ranking. The method should therefore be able to select different loops by global exploration of conformational space independently of starting conditions. Methods building the loop stepwise from one anchor residue to the other bias the solutions depending on choices made in conformation of the first few residues. Rather a global approach to the optimization is required.

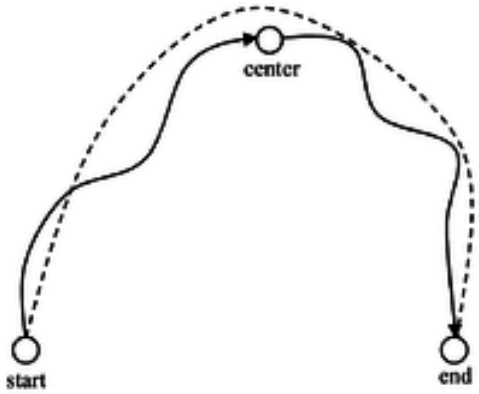


Figure 39 Lobo loop modeling

4. Results

4.1. ReUPred performance

ReUpred (L. Hirsh et al. 2016) is a predictor based on the structure of repeat units and their structural alignment. We looked for a similar application in order to have a benchmark but to our knowledge there is no similar tool. Instead we found two different applications that predict repeat units but using different approaches. These applications were: TAPO: A combined method for the identification of tandem repeats in protein structures (Do Viet, Roche, and Kajava 2015) and ConSole: using modularity of contact maps to locate solenoid domains in protein structures (Hrabe and Godzik 2014). We used Raphael (Walsh et al. 2012) dataset in order to create a benchmark, it contained 105 known solenoids structures and 247 non-solenoids structures. To do the analysis we considered separately each of the three types of solenoids (α, β and α/β) as well as all together. And as shown Figure 40, all Precision, Recall, FMeasure and accuracy are higher for ReUPred than for Console or TAPO.

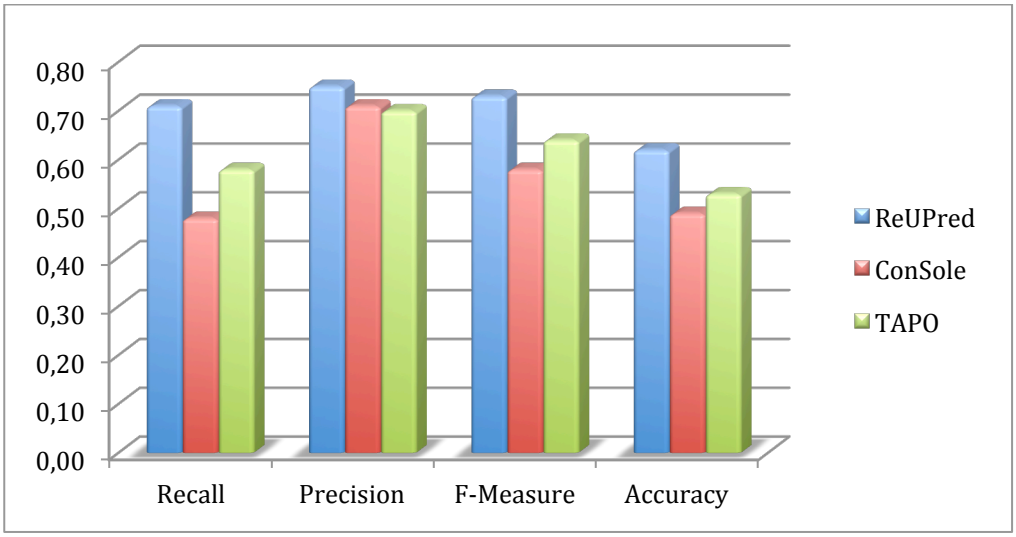


Figure 40 ReUPred versus TAPO and Console

We also evaluated the quality of the predicted units obtained with each of the applications and considered all solenoid types then compared the resulting values with units found in RepeatsDB. From observations we can state that ReUPred's performance for detection of unit lengths has proved better than TAPO's and Console's (Figure 41).

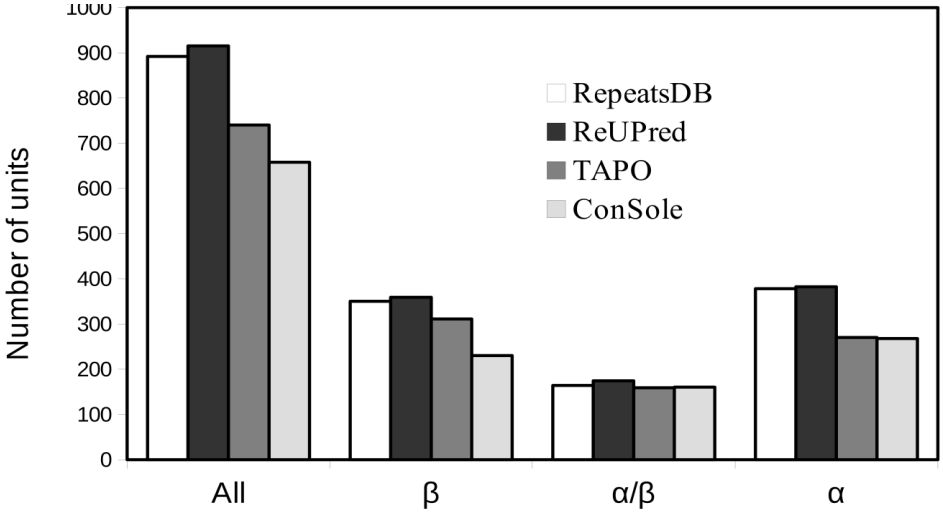


Figure 41 Number of units predicted by ReUPred, TAPO and Console

Then using data gathered from RepeatsDB, we evaluated the length of all units based in the number of residues that conform it. And as shown in Figure 42, ReUPred is the most similar to the original data.

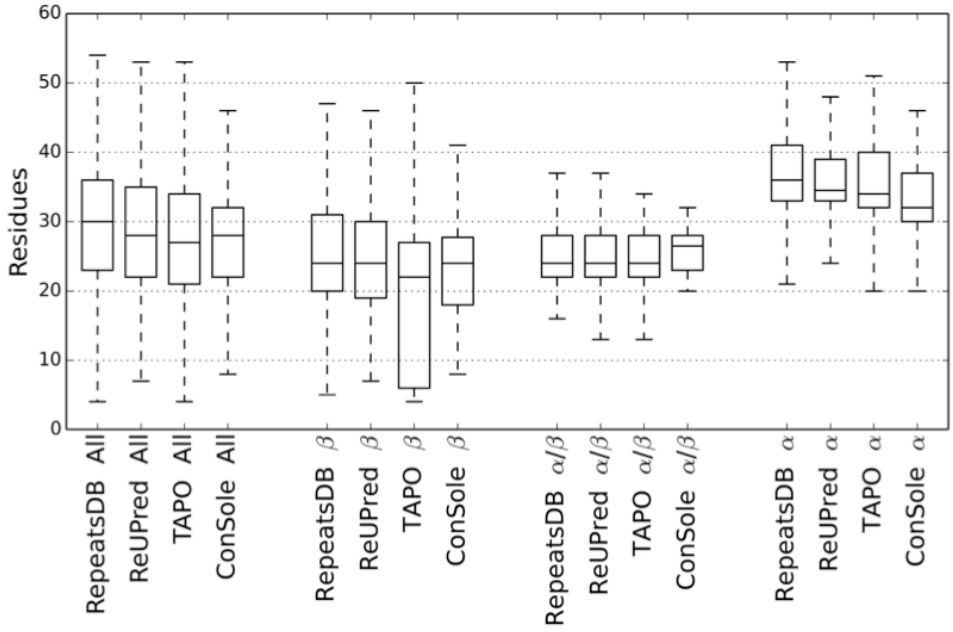


Figure 42 Predicted units by ReUPred, TAPO and Console

After elaborating a benchmark of ReUPred against the other two applications, we evaluated ReUPred in a more detailed way, analyzing the predictor performance by subclasses: α, β, α/β as well as all together. ReUPred algorithm, as explained before, is based on a template unit allocated in Structural Repeat Unit Library (SRUL), and this unit is related to a specific

subclass. However, information used to "identify" the subclass of a target protein in the case of α/β-solenoid template units sometimes could result as being more structurally similar to an α-solenoids or more structurally similar to α/β-solenoid. This is the reason why mixed α/β-solenoids have lower values than α-solenoids and β-solenoids, as shown in Figure 43. We also noted that α-solenoid is the subclass in which ReUPred has almost a perfect performance in classifying target protein. The reason for this is more about regularity on secondary structure of unit inside this subclass. While in the case of β-solenoids, we have units of different length and a lot of loops/insertions.

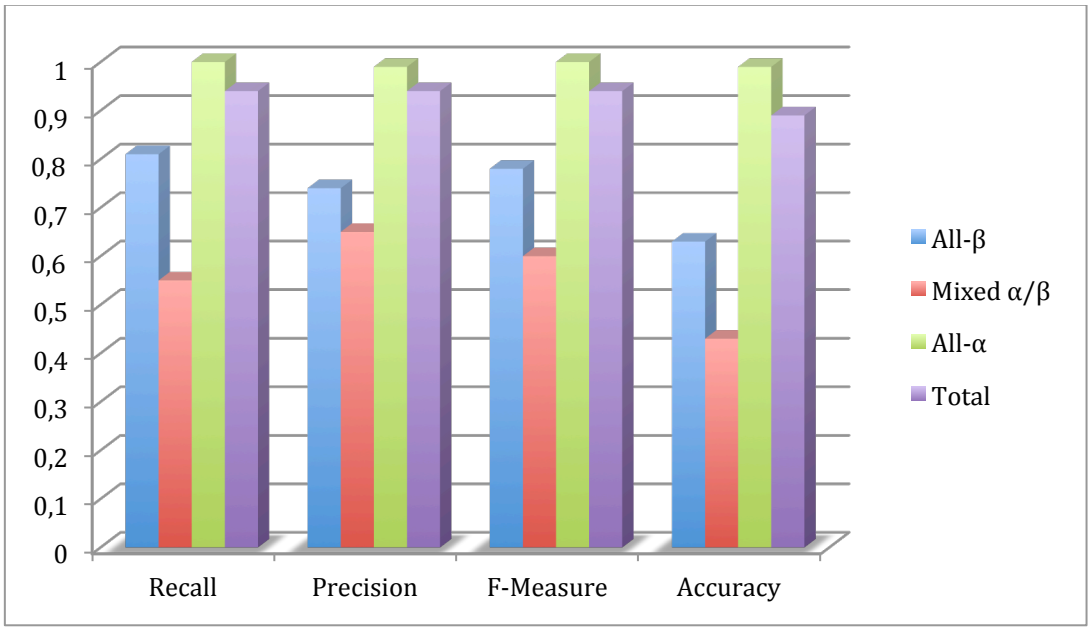


Figure 43 ReUPred performance for solenoid subclasses classification

Inside RepeatsDB we were able to find a period for each of the entries used in dataset, this period value was obtained using Raphael (Walsh et al. 2012), an application that determines period in a protein structure and it is able to identify if a protein is repeated or not. So to evaluate if ReUPred had similar values to the ones calculated by Raphael, we obtained the average length of predicted units and compared resulting values with the period obtained from the database (Figure 44).

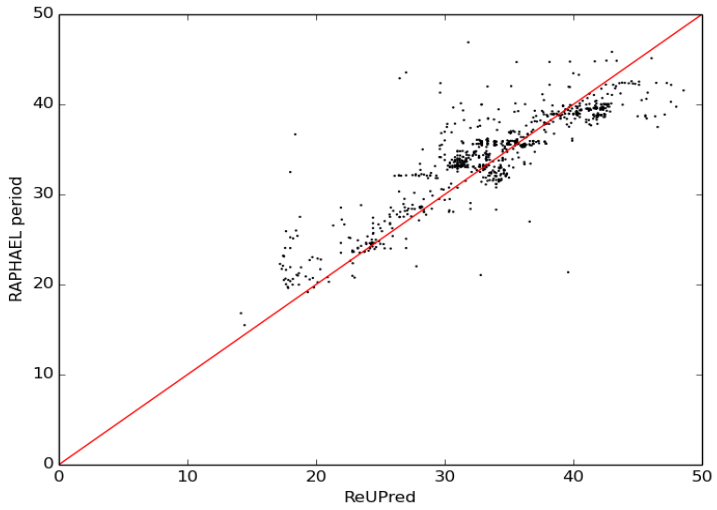


Figure 44 Period from RaPhael versus average unit length of ReUpred

Then we replicated the analysis but instead of using Raphael period against the average length, we compared values with the lengths of predicted units. And as shown in Figure 45, the values were almost the same, which again confirmed the good performance of ReUPred.

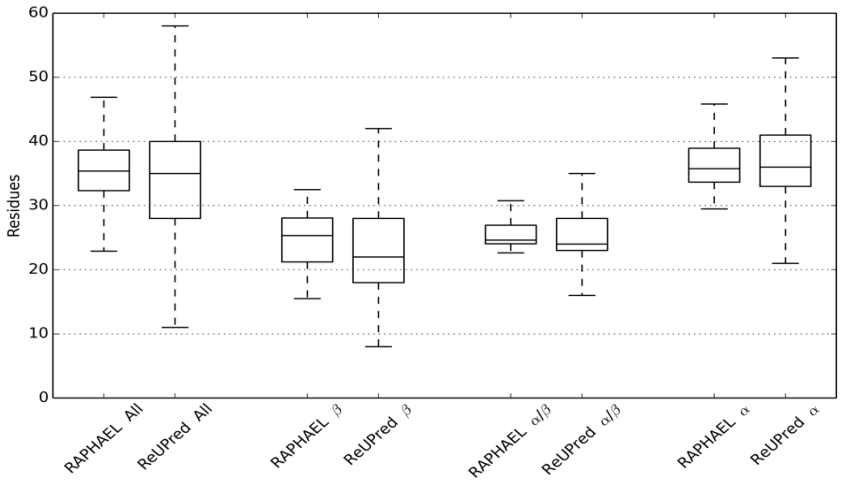


Figure 45 Predicted units by ReUPred, TAPO and Console

Finally we decided to run the predictor with a different dataset of 1075 classified and by similarity known repeat proteins from RepeatsDB, as we did not have information about repeat units to make a similar analisys as before, we mapped

predicted units with PFAM's information of proteins. Based on our observation we can state that ReUPred is able to increase annotation by an order of magnitude for both of the datasets (Figure 46).

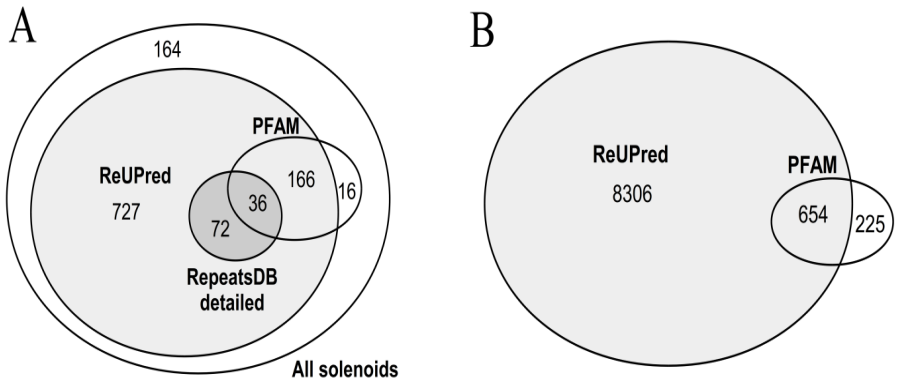


Figure 46 Venn diagram of available annotation for RepeatsDB classified dataset

4.2. RepeatsDB content and structural classification

RepeatsDB is a database that contains information about repeat proteins (Di Domenico et al. 2014a). The second version, RepeatsDB 2.0 contains a total of 5474 entries with detailed information predicted by an optimization of ReUPred (L. Hirsh et al. 2016) and 3307(65%) of these entries were manually curated. As shown in Table 8 all these entries correspond to 970 UniProtKB different sequences (Paladin et al. 2016) .

Table 8 Statistics by class of RepeatsDB 2016

Class	Chains		UniProt
	Reviewed	All	All
II	12	12	4
III	1356	2367	405
IV	1777	2780	486
V	162	215	75
TOTAL	3307	5374	970

This database update presents more information for each entry (Figure 47). In the top part of the page it reports structure information from PDB and cross-references to third party databases (UniProtKB, MobiDB, SCOP, CATH and Pfam). Then it displays a table in where region details are shown (structural

classification, start end, number of units, periods and cluster families). In the following a section feature viewer, summarizes available annotation for PDB reference sequence, i.e. the SEQRES field in PDB file. An overview of RepeatsDB information (regions, units and insertions) along with secondary structure (DSSP), Pfam, SCOP and CATH tracks (when available) are also shown. Finally in the bottom section a detailed view of RepeatsDB annotations is highlighted in sequence and PDB viewers.

As a result of ReUPred execution on all PDB databases, we were also able to identify new structural subclasses on class IV (Closed structures whose repeat units need one another to maintain structure) and class V (Beads on a string' structures whose repeat units are large enough to fold independently) that were included in the new version of the database.

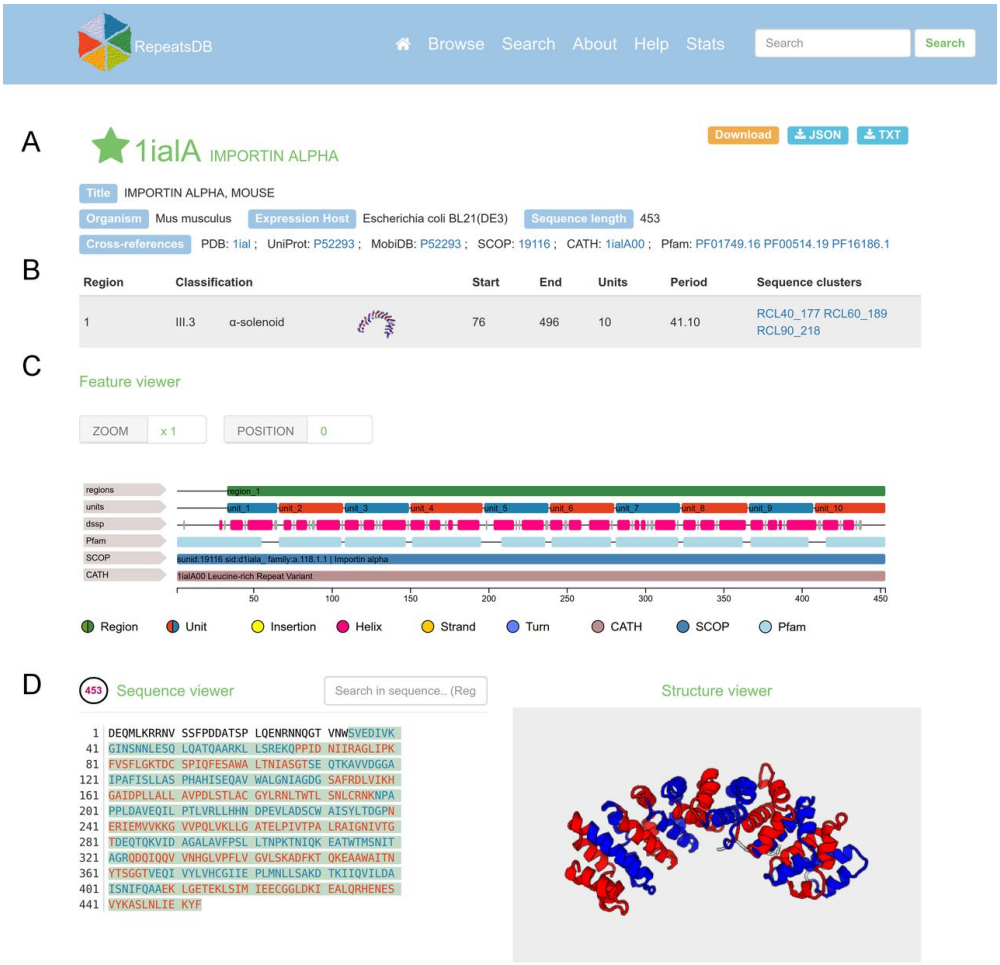


Figure 47 RepeatsDB 2,0 sample entry

These new subclasses are (Figure 48 a-e):

For class IV:

- α/β barrel (IV.7), always a five α/β unit closed structure .
- α/β propeller (IV.8), similar to a β propeller in shape and number but with the presence of α helix inside the unit.
- α/β trefoil (IV.9), similar to a β trefoil in shape and number but with the presence of α helix inside the unit.

For the class V:

- α/β sandwich (V.4), each sandwich bead is formed by "two layers" of α/β.
- β sandwich (V.5), same as α/β sandwich but in the unit there are only β strains.

Two later identified subclasses are aligned prism (Figure 48f) and box (Figure 48g), these two new subclasses are not included in the current version of RepeatsDB, but will be included in next update. The reason for not including them was not only that we realized their existence at the last minute as is the case of the aligned prism but also the uncertainty of its class as is the case of the box. It seems to be an elongated structure but it closes in the presence of enough repeat units (Figure 49) and this is the reason why we were unable to define its right class.

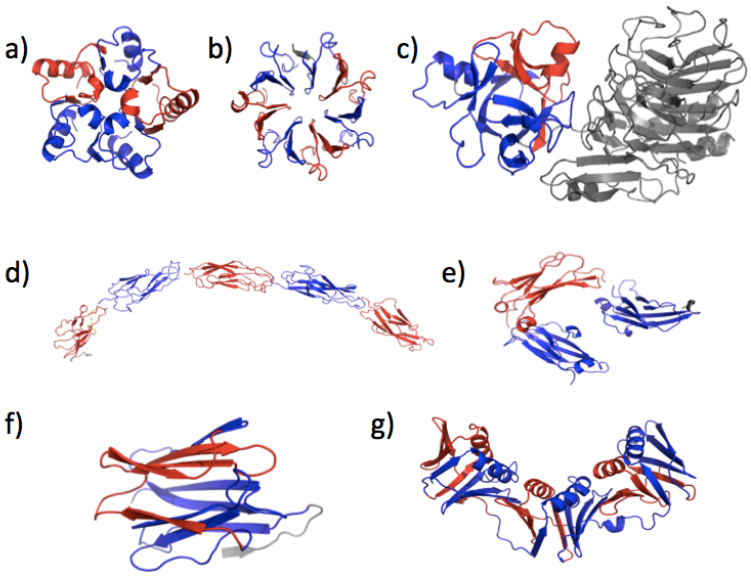


Figure 48 a) α/β barrel 1g61A, IV.7 sample b) α/β propeller 3qi0B, IV.9 sample c) α/β Trefoil 2d43A, IV.9 sample d) β sandwich bead 1q55C,V.4 sample e) α/β sandwich bead 2wqrB ,V.5 sample f) Align prism 3wocB, IV.10 sample g) Box 2xurA, III.7 sample

The problem with the box structure made us think if a new classification might be needed, also because there are some cases in which a elongated structure can became a closed one if the right number of units is reached, as is the case of some α/β-solenoids.

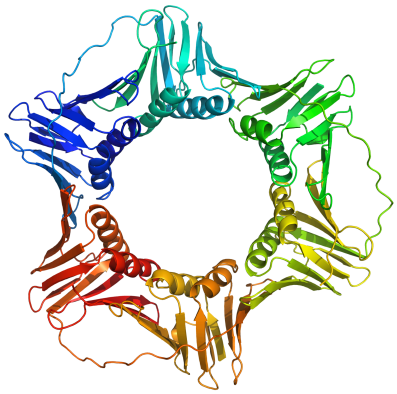


Figure 49 Closed Box 3k4xA

In 2012 Kajava proposed a structural classification representation, in were we could identify subclasses based on unit length. In Figure 50, we can see a similar representation but it includes new subclasses present in RepeatsDB 2.0. As we can see in the figure and comparing length values with the ones stipulated by Kajava, the average length of repeat units defined before are pretty similar to the ones presented in RepeatsDB 2.0 (Table 9), which allowed us to confirm the values accuracy (Andrey V. Kajava 2012a)

Table 9 Comparison of repeat unit length of 2012 and 2016 by class

Residues in the repeat unit	**2012**	**2016**
Elongated Structures (III)	5 to 45 residues	21 to 37
Closed structures (IV)	35 to 65	26 to 65
Beads on String (V)	30 to 60 / 100 - 130	31 to 100

In class V, we can observe two particular cases, subclass V.3 (α/β-beads) that has a shorter average length of 31.36 with respect to the estipulate by Kajava. And also in subclass IV.5 (α/β-prism) we see a unit length of 65.16, which is also out of the estipulate length possible values. We hypothesized that these results were not in the previous ranges because of the new structures and therefore not observed in 2012, and not because of a mistake in observations.

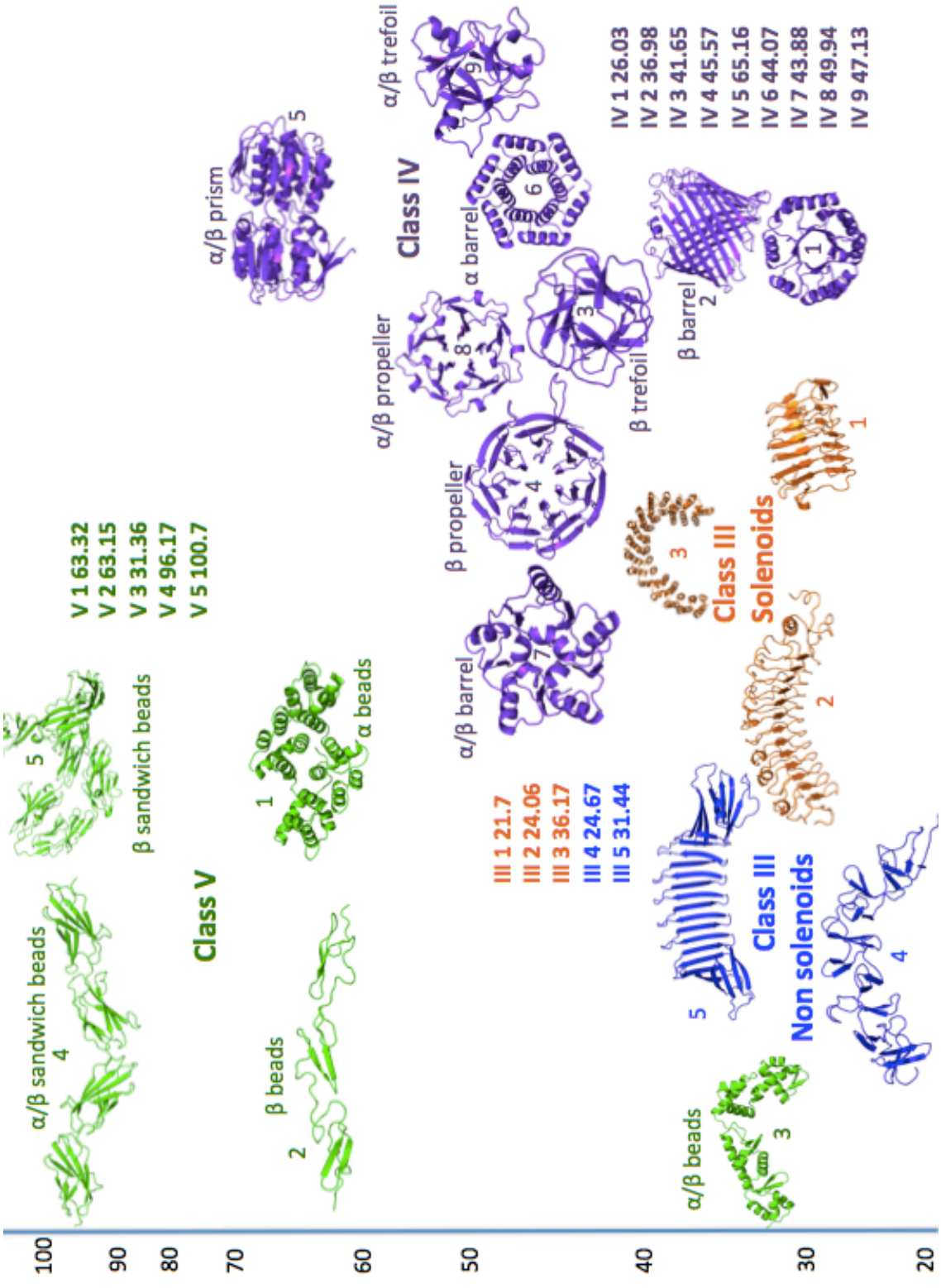


Figure 50 Representation of structural classification of repeatsDB 2.0

4.3. α-Solenoids ensembles

After filtering and curating processes (see methods 4.3), we got 740 units and aligned them structurally using TMalign (Zhang Y and Skolnick J. 2005). Then

we created a new network of 740 nodes shown in Figure 51 in where each node represents a repeat unit and each edge is the TM-score value. In it we used a value of 0.7 of TM-score as threshold and we were able to identify 7 clusters, each related to a specific unit structure. Unit clusters are colored by PFAM assignments showing an almost perfect separation. The nodes in grey are ones in which no Pfam code is assigned (not recognized as part of a known family).

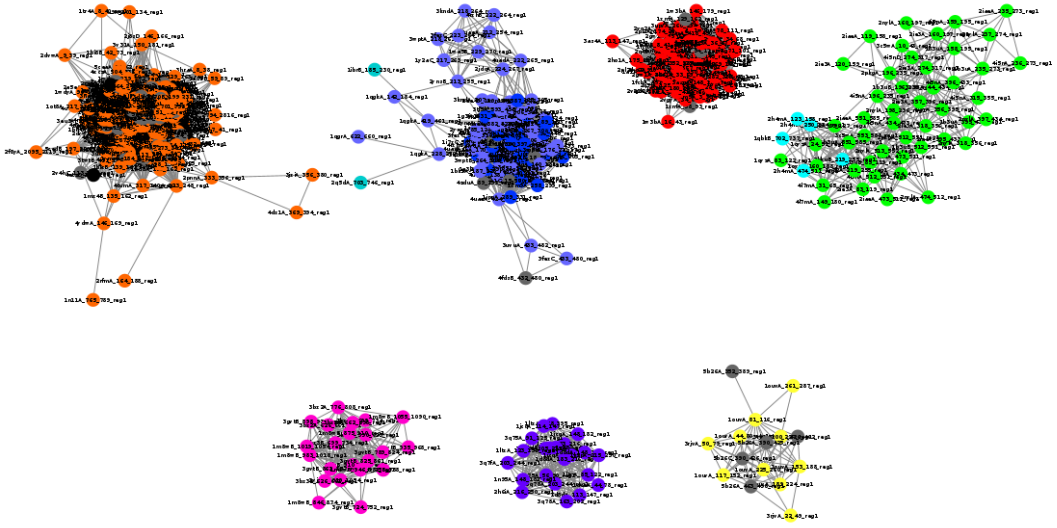


Figure 51 Network based on the 613x613 matrix

Looking at clusters in the network we were able to relate each cluster with a different type of structure, Figure 52. There are four clusters related with a specific conformation, while there are three mixed clusters, in two of them the presence of Importin is noticed. Importin is the only family that does not form its own cluster but is mixed only with HEAT or with Armadillo. In the Ankyrin cluster different sub-families coexist.

We took all units inside each of the clusters and analyzed them separately. In each of the clusters we calculated the multiple structural alignment using Mustang-MR structural Sieving Server (Konagurthu et al. 2006). For each structural alignment we exploited the corresponding sequence alignment to build a Hidden Markov Model (HMM) by using HMMer (R. D. Finn, Clements, and Eddy 2011). Then we also created the corresponding sequence logo (Crooks 2004) and calculated the secondary structure consensus. We started our analysis with a cluster with only one Pfam family: SEL (yellow), PPTA (purple), Pumilio (magenta) and TPR (red). To calculate the secondary

structure, we took all the aligned units inside each of the clusters and using DSSP (Touw et al. 2015) in each of them we found a secondary structure consensus using a threshold of 70% of the units in the clusters in order to decide what was the secondary structure of each of the residues. To do the analysis we decided to use The Residue Interaction Network Generator (Ring), (Piovesan, Minervini, and Tosatto 2016), and with it identified interactions at an atomic level in the structures.

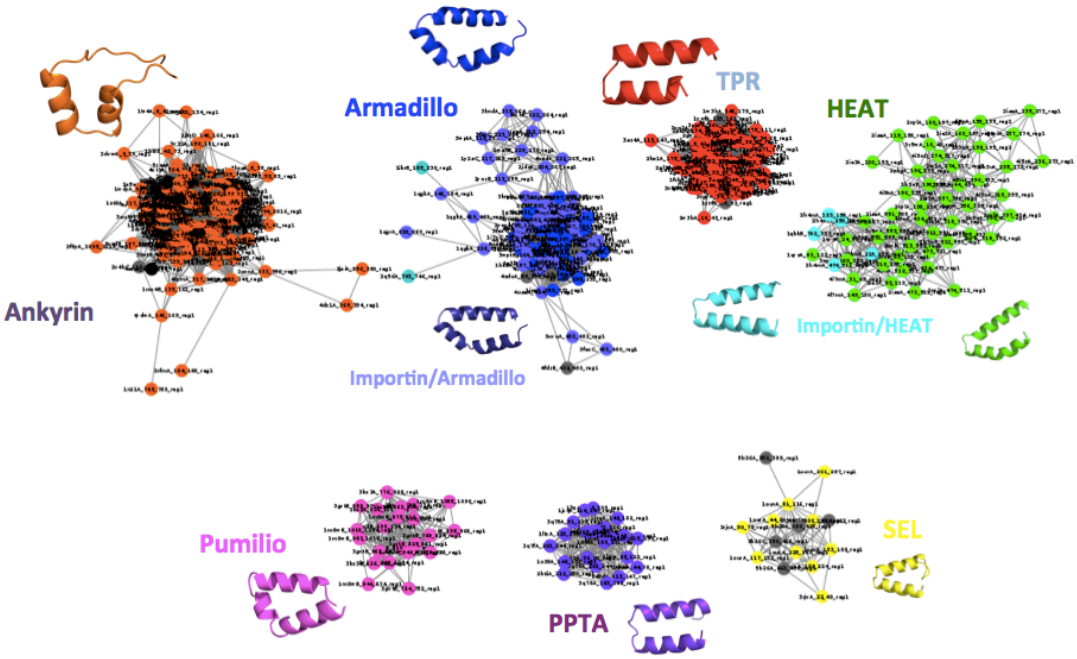


Figure 52 Ensembles with their corresponding unit structure

4.3.1. SEL1 (PF08238)

Pfam defines SEL1 family (PF08238) as TPR clan (CL0020) member that is mostly found across eukaryote and bacteria species, its HMM logo has a length of 38 residues. InterPro (Robert D. Finn et al. 2017) identifies Sel1-like repeats (IPR006597) as tetratricopeptide repeat sequences originally identified in Caenorhabditis eleagans receptor molecule, which is the key negative regulator of the Notch pathway.

Our secondary structure consensus model shows (Figure 53) that their units are formed by two α-helices, with a hydrogen bond in the middle and a bend near the C terminal. In the HMM Logo (Figure 53) we can observe how alanine (A), an aliphatic and hydrophobic residue is almost always present in the third position of the first α-helix. It seems that together with glycine (G) and leucine (Leu), present on the middle of the second helix, it contributes to stabilize the protein fold. In Figure 54 we show the conserved residues in color green and black, considering a 55% of sequence identity the conserved residue are the previously mentioned. We can also observe in the figure how these conserved residues seem to give the protein structure a specific twist. However in order to prove our hypothesis we might need to do mutagenesis experiments.

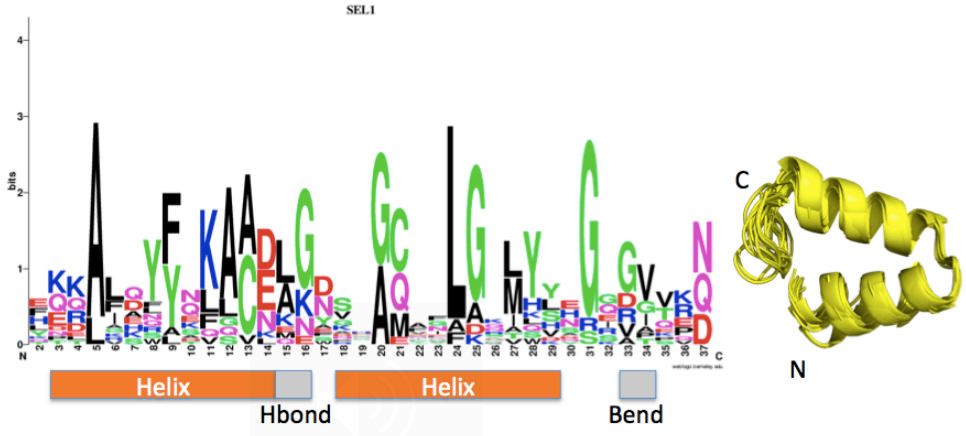


Figure 53 SEL1 logo representation and units structures aligned

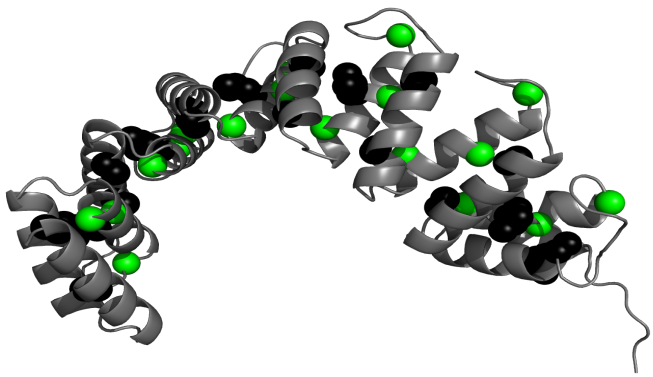


Figure 54 SEL1 1ouvA structure showing mostly conserved residues in the logo(position 5, 12, 16, 24,25,31)

We decided to take one particular protein of the cluster, 1OUV chain A (PDB Helicobacter cysteine rich protein C) and align all its repeat units using Jalview (Konagurthu et al. 2006). In this specific case we observe

how with a sequence identity of 55% we have even more conserved residues (Figure 55). If we see all these residues in the structure we will see how a lysine (K) positively charged together with a negatively charged residue aspartic Acid (D), seem to be interacting (Figure 56, Figure 57, Figure 56).

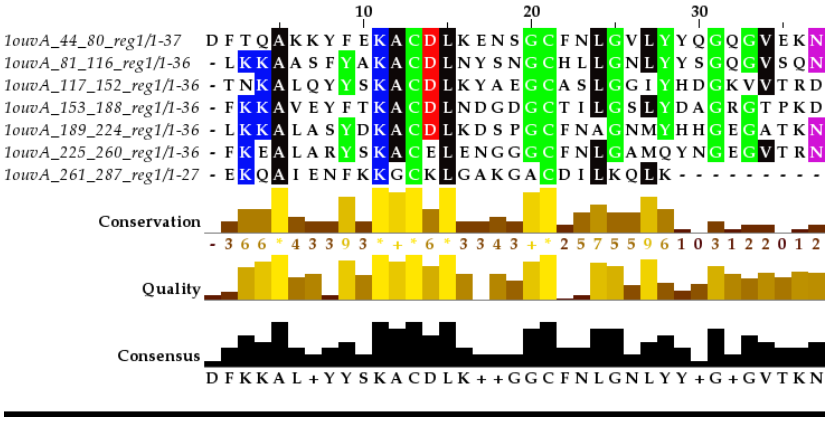


Figure 55 SEL1 1ouvA sequence alignment of structural units

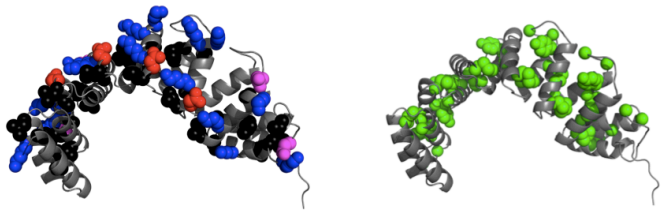


Figure 56 SEL1 1ouvA structure showing most conserved residues

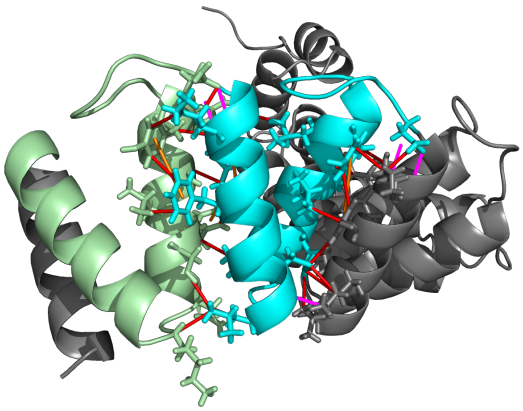


Figure 57 SEL1 1ouvA structure showing interactions between conserved residues between two units

Inside this cluster the shorter unit has 27 residues and the longest unit has 37 residues. The average length is 34,4 considering a total of 17 different units that form SEL1 cluster. Values coherent with the ones

from Pfam that have a 38 residues long and three of those residues have a 25% probability of being part of an insertion.

4.3.2. PPTA (PF01239)

PPTA family (PF01239), protein prenyltransferase α subunit, is defined by PFAM as a member of TPR clan (CL0020) and it is mostly present in eukariota. This family corresponds to an InterPro entry defined as "posttranslational attachment of either a farnesyl or a geranylgeranyl group via thioether linkage to a cysteine or near carboxyl terminus of the protein" (IPR002088). Its secondary structure consensus shows that its units are usually formed by two α helices and in the middle there is a hydrogen bond. In PPTA Logo (Figure 58), we can observe a high quantity of leucine (L) all around the repeat unit. We also notice presence of other four residues highly represented: glutamic acid (E). One position before the beginning of the second helix, we can also observe asparagine (N) a polar residue. The tryptophan, the largest among amino acids, is highly present in the second α helix. Finally the arginine (R), a positively charged residue is also present in the middle of the second α helix.

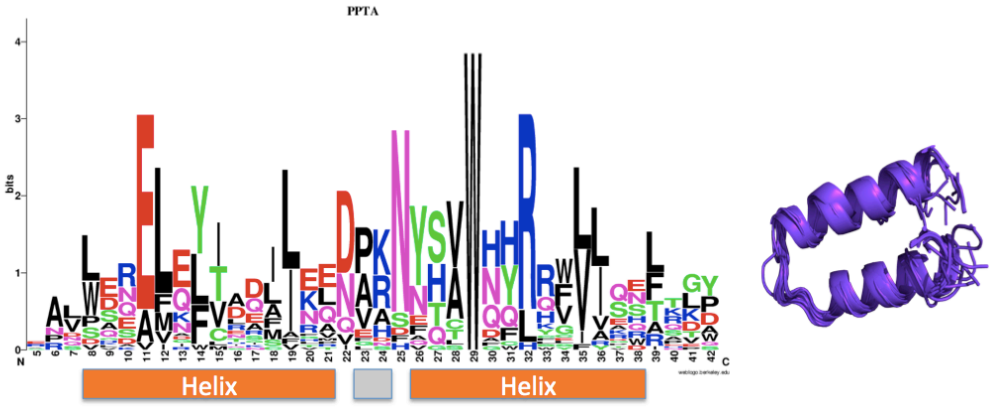


Figure 58 PPTA logo representation and units structures aligned

Again we observe a particular protein 1JCQ chain A (PDB Crystal structure of Human Protein farnesyltransferase complex with farnesil diphosphate and the peptidomimetic inhibitor L-739,750) of the cluster to see if conserved residues interactions were the reason for the folding type and looking at Figure 59 we can confirm how interactions between

conserved residues of the unit seems to be responsible for this particular twist of the protein (Figure 60).

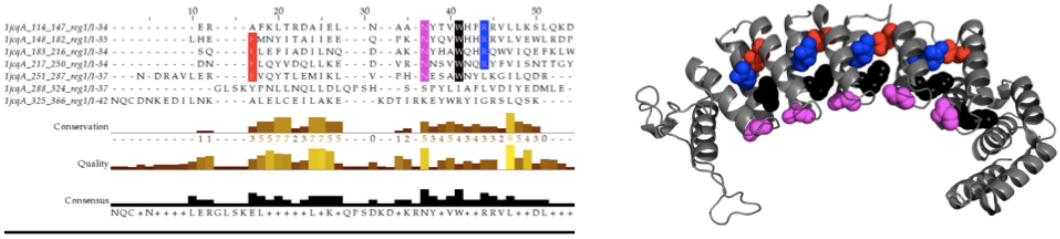


Figure 59 PPTA 1jcqA sequence alignment considering an identity of 55 and the structure showing the conserved residues

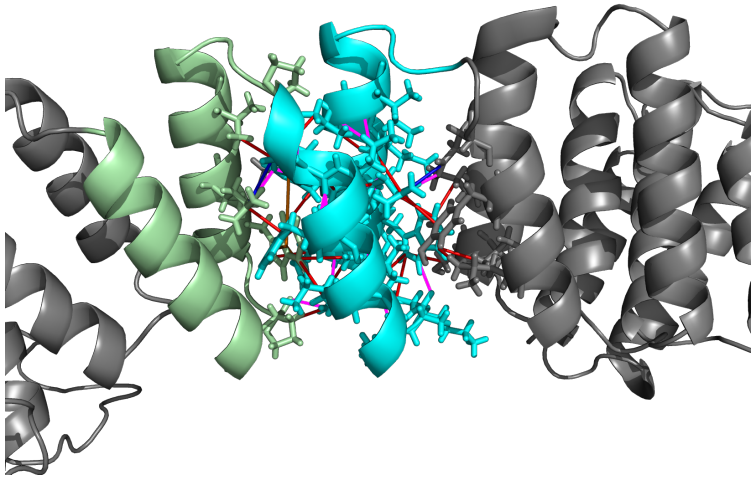


Figure 60 PPTA 1jcqA structure showing interactions between conserved residues between two units

Inside this group, the shorter unit has 32 residues while the largest has a length of 41. Average length is 35.1, considering all 29 units of the group. This average length is pretty different to the length in Pfam, 28 residues long. The reason for this difference is the definition of the repeat unit; our first six residues are ignored by Pfam and ignored as a part of their unit. Comparing our values to those of SEL1, we could say that even that both repeat units are composed by two α helices, each one has a different length and conformation, in where repeat units of PPTA are longer than units of SEL1.

4.3.3. Pumilio (PF00806)

Pfam defines Pumilio family (PF00806) binding repeat (Puf repeats) as necessary and sufficient for sequence specific RNA binding in proteins

(fly Pumilio and worm FBF-1 and FBF2) that function as translational repressors in early embryonic development by binding sequences. In this particular case the logo (Figure 61) does not show any particular conserved position, this is why we also present the sequence alignment of the cluster units to have a better level of detail (Figure 62). The secondary structure shows three α helices, two of a longer length in the N-terminus and C-terminus and a smaller one in the middle.

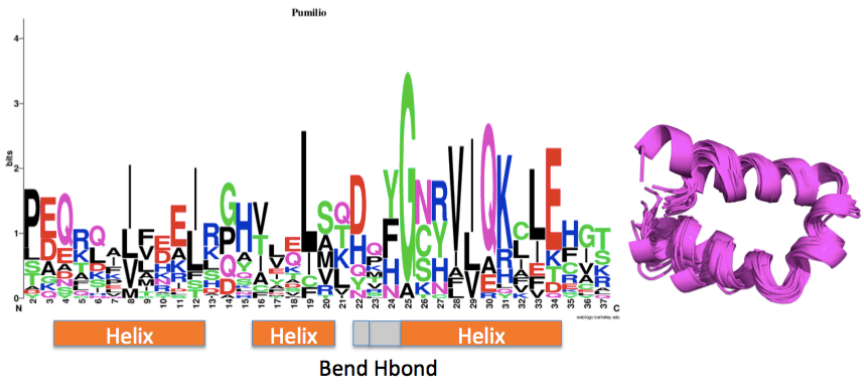


Figure 61 Pumilio logo representation and units structures aligned

This family is also part of TPR clan (CL0020) and is usually found in eukariota. The alignment shows the presence of not only leucine (L) but also Isoleucine and valine (V) around repeat units, three amino acids that have hydrocarbon side chains. We also observe that Glycine (G) an aliphatic residue is highly present as is the case of glutamine (Q) an uncharged residue and glutamic acid (E), a charged residue. All this might explain why the protein has this twist and its units seem to be almost parallel. We also observed a particular protein of this family, 3GVT chain B (Structure and RNA binding of the mouse Pumilio-2 Puf domain), and in the structure we can see how conserved residues are in the inner and outer parts of the protein. This is probably the reason why there is this curve in the structure and why the units seem to be almost parallel between each other.

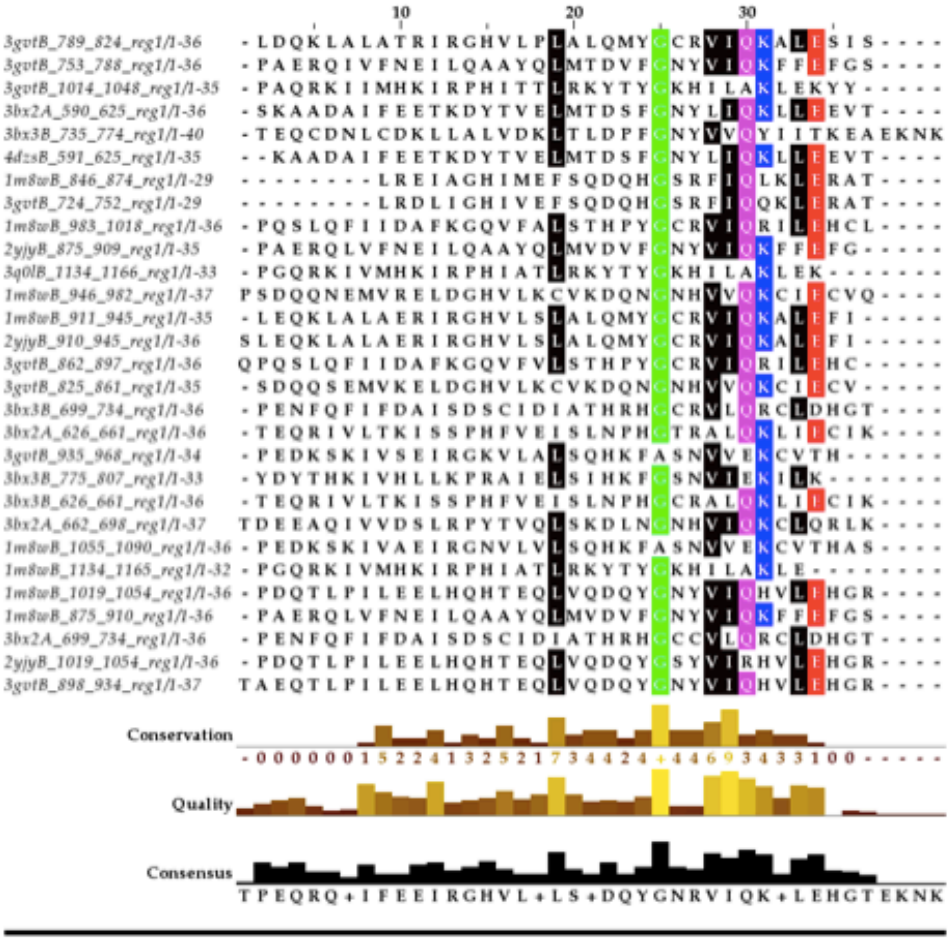


Figure 62 Pumilio sequence alignment showing an identity of 55

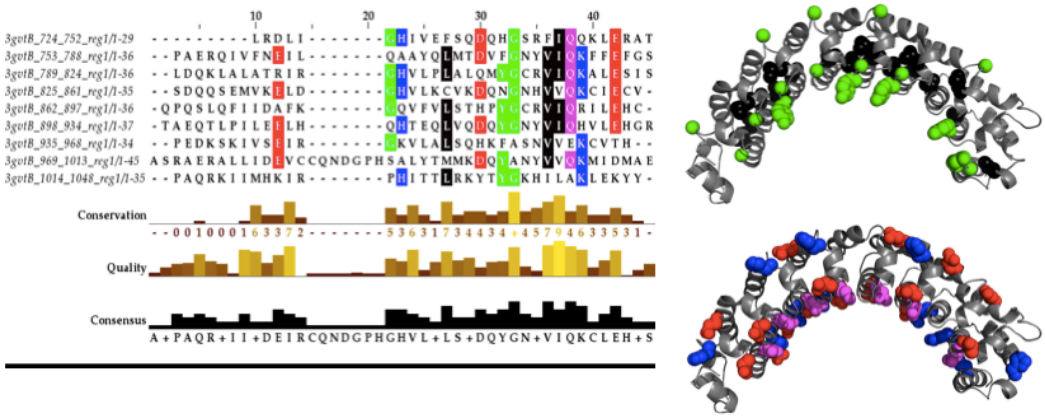


Figure 63 Pumilio 3gvtA sequence and structure with conserved residues

Looking at the same structure but only coloring the residues conserved in the logo Figure 64 we can see how all these residues interactions are the ones causing the semi closure of the structure. In this group we have 29 units, from which the shortest has 29 residues and the longest has 40 residues and average length of 35.1 residues, same value of PPTA unit

average length but with a different number of α helixes. Pfam presents a length of 35 residues length, almost the same than our results.

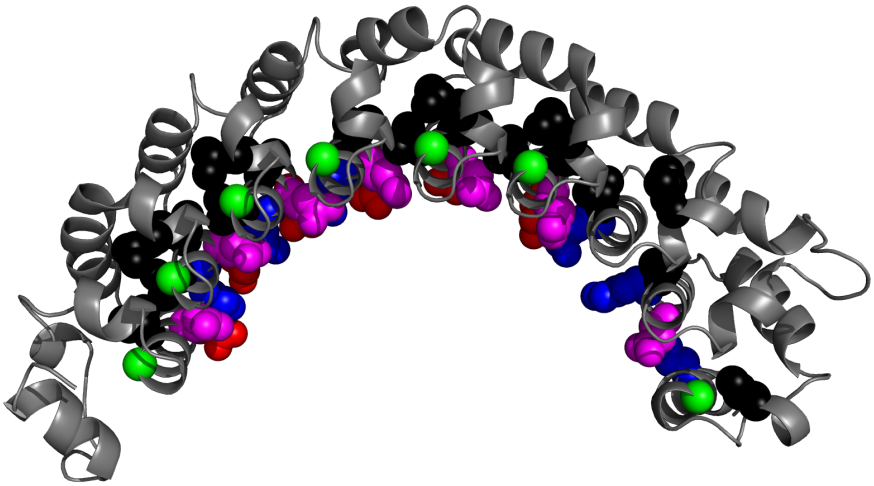


Figure 64 Pumilio 3gvtA structure showing conserved residues of the logo

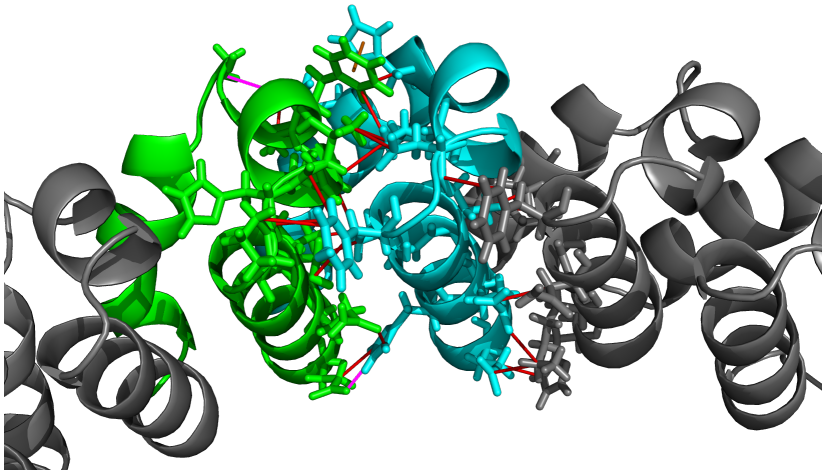


Figure 65 Pumilio 3gvtA structure showing interactions between conserved residues between two units

4.3.4. TPR (PF07719)

Finally, Pfam has 21 different Tetratricopeptide repeat families, we based our observations using TPR_2 (PF07719) because of the resemblance with our logo and also because it is the one with less insertions. InterPro defines this entry (IPR013105) as: " a structural motif present in a wide range of proteins. It mediates protein-protein interactions and the assembly of multiprotein complexes ".

Observing our logo (Figure 66) we can say that there are many aliphatic amino acids as is the case of alanine (A) that is highly conserved at the beginning of the second helix. Also present all around the logo are leucine (L), another aliphatic non-polar amino acid, together with the glycine (G). Another highly conserved amino acid present at the end of the logo is the proline (P). In the TPR families from Pfam we can observe almost the same conserved residues than in our Logo, the differences are in the level of conservation of the residues and the possible insertions and deletions, which affects the length.

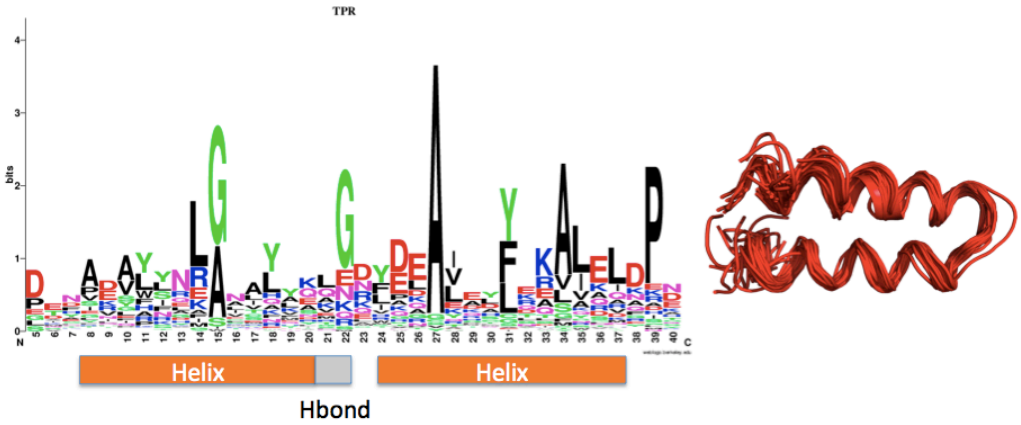


Figure 66 TPR logo representation and units structures aligned

We can see that the unit is formed by two α helices and in the middle there is a hydrogen bond, pretty similar to SEL1 units but with out the bend at the end. For this family we evaluated 1W3B (PDB: Superhelical TPR domain of O-linked GLCNAC transferase reveals structural similarities to importin alpha) protein chain A and we identified some conserved residues (Figure 67) that seem to be responsible for the protein structure fold, as shown in Figure 68. Furthermore it appears that the conserved residues give a particular twist to the protein structure (Figure 69). In these clusters we found more diversity in the units so we have a total of 89 from which the shortest has a length of 28 residues and the longest has 37 residues. Average length of the units is 33,76 residues, the smallest value comparing to the previously mentioned average lengths.

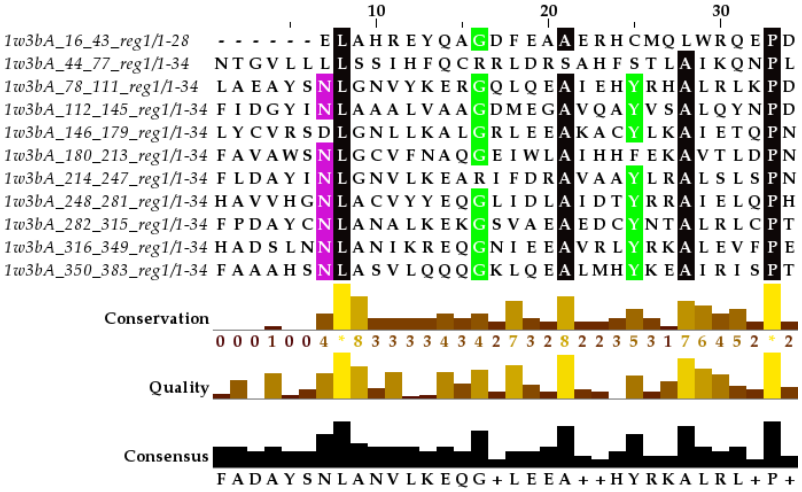


Figure 67 TPR sequence alignment of 1w3bA repeat units

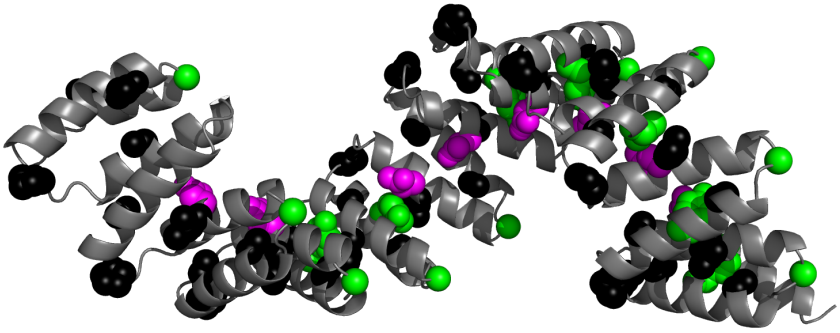


Figure 68 TPR 1w3bA structure with the conserved residues marked

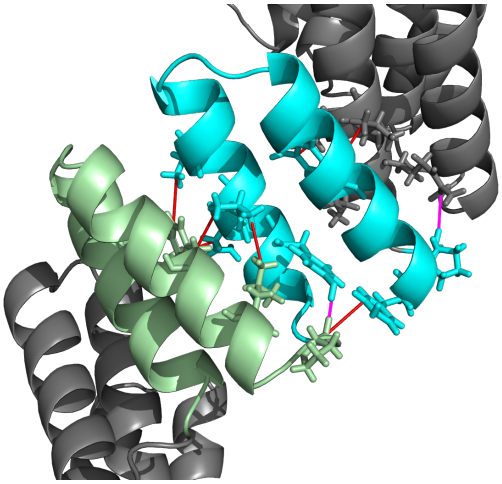


Figure 69 TPR 1w3bA structure showing interactions between conserved residues between two units

SEL1, PPTA, Pumilio and TPR are the four simplest clusters of our network. All their average unit lengths are inside a range of 33.7 to 35.1 residues. And while Pumilio has three α-helices inside the repeat unit, all the others have only two. The complete structure of PPTA and Pumilio seem to have the same arc shape.

In the units that have no twist as is the case of TPR and SEL1, we can observe a twist in the whole structure.

In the network (Figure 52) there are three more complex clusters. Armadillo and HEAT are both mixed with Importin, while Ankyrin cluster has some nodes inside marked by Pfam as well as different related famililies. Some of this classes are Adeno knob (PF00541), CC2-LZ (PF16516), Peptidase_C14(PF00656), among others.

4.3.5. Ankyrin

Ankyrin repeat family (PF00023) is a 33 residues motif in proteins and one of the most commonly known. It consists in two α helices separated by loops; it was discovered at first in signaling proteins in yeast and Drosofphila Notch. Domains of this type mediate protein-protein interactions. They are usually present in bacteria, archaeal and mostly in eukaryotic proteins. Pfam states that this family is member of clan Ank (CL0465) together with other four ankyrin repeats. In this case, we use structural alignment of the units to see the differences inside the cluster (Figure 70) and we were able to observe that the differences are mostly in the N-terminal and C-terminal regions. This observation is confirmed by looking at the corresponding sequence alignment (Figure 71) where we can see that gaps are present at N and C terminal regions and also that in the last section there are two single gaps present. As we preferred a perfect alignment without internal gaps for our analysis, we decided to work separately on this cluster, increasing TM-score threshold.

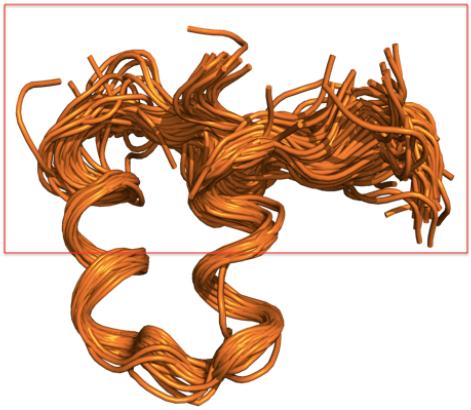


Figure 70 Not perfect structure alignment of Ankyrin units

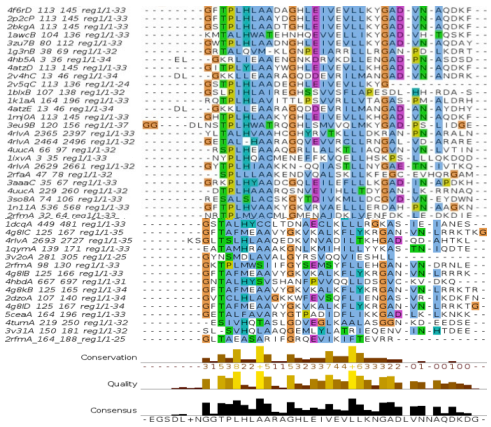


Figure 71 Not perfect sequence alignment of Ankyrin units

Then using a TM-score value higher or equal than 0.91, we obtained two different clusters, one containing all the defined by PFAM such as ankyrin (Figure 72) and the other containing a mix of different families with also some ankyrin units (Figure 73).

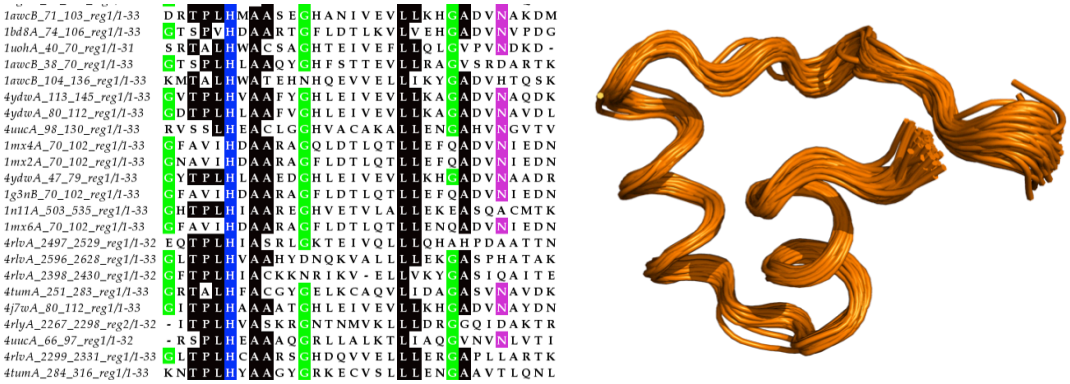


Figure 72 Ankyrin sub cluster 1 sequence and structural alignment

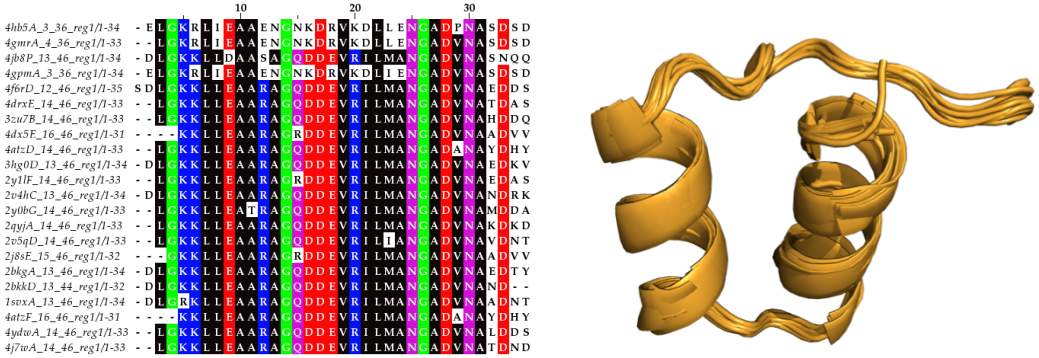


Figure 73 Ankyrin sub cluster 2 sequence and structural alignment

Between sub-clusters there are observable similarities and differences but in general the glutamine (Q) is one of the residues mostly present in the second sub cluster, together with lysine (K), arginine (R) and aspartic acid (D) (Figure 74). While in the first sub cluster, proline is mostly present together with the histidine (H) (Figure 75).

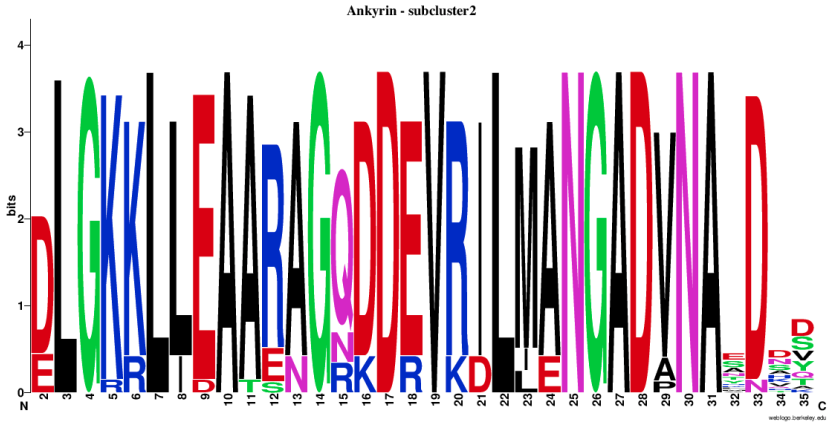


Figure 74 Ankyrin Logo of sub cluster 2

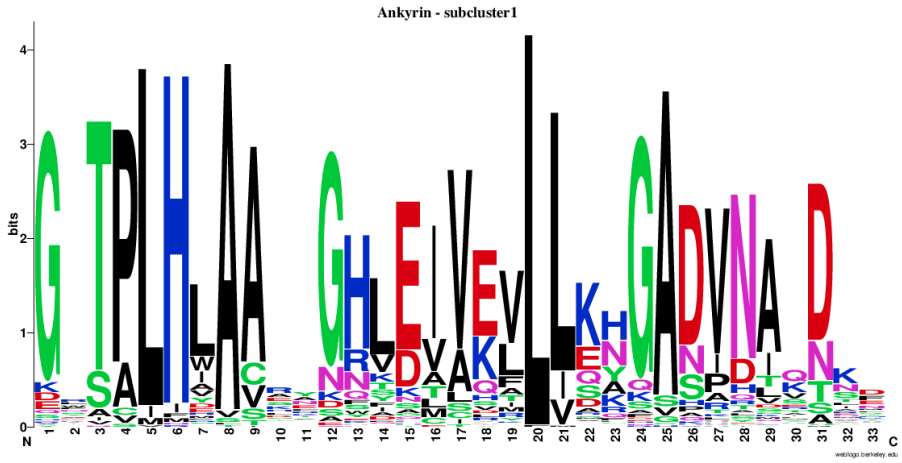


Figure 75 Ankyrin Logo of subcluster 1

For our first sub cluster, the one containing only ankyrin or undefined units, we decided to replicate our analysis using 3IXE chain A protein (PDBStructural basis of competition between PINCH1 and PINCH2 for binding to the ankyrin repeat domain of integrin-linked kinase). In it we can see how residues are aligned (Figure 76;Figure 77) and how they interact (Figure 78). For the cluster we have an average length of 34 residues, while Pfam has a 32 residue length for ankyrin family and considers many possible insertions. In general the HMM logo of the sub cluster 1 is almost the same than the Pfam ankyrin HMM logo.

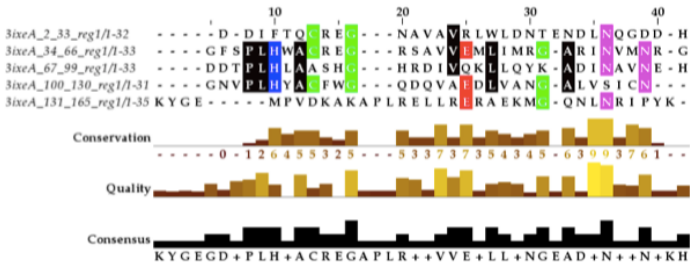


Figure 76 Ankyrin 3ixeA protein alignment

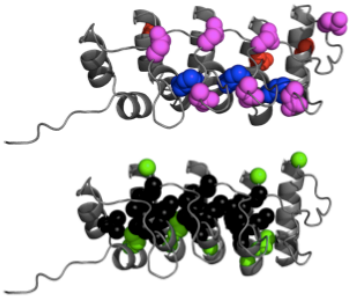


Figure 77 Ankyrin 3ixeA protein structure

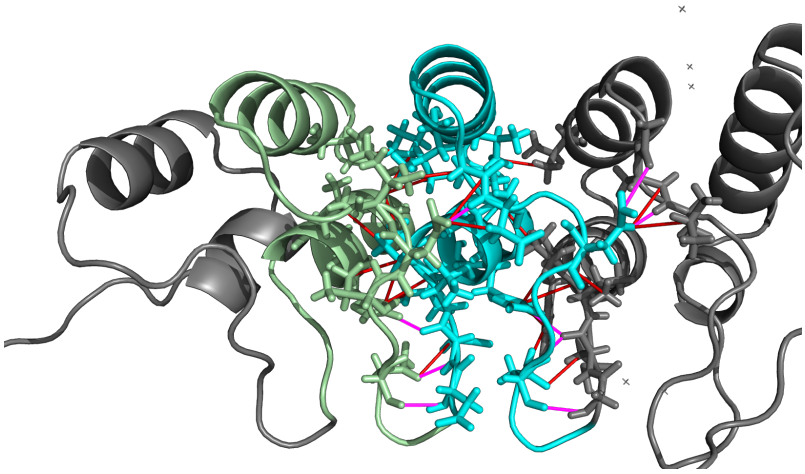


Figure 78 Ankyrin 3ixeA protein interactions

4.3.6. Armadillo

Another cluster with mixed families is the Armadillo/β catenini like repeat (PF00514). Repeats of this family are usually a 40 amino acid repeated sequence motif. It was first identified in drosophila melanogaster segment (InterPro domain IPR000225). Pfam presents 3 families of armadillo: Armadillo/β catenin like repeat (Arm), Armadillo like (Arm_2,

PF04826), Atypical Arm repeat (Arm_3, PF16186) and another set for ARM-DNA-bind. An interesting observation in this case is the Atypical Armadillo repeat. Pfam states that: “this kind of repeat appears at the very C-terminus of eukaryotic proteins such as importin subunit α2, as the last of the repeating unit” and InterPro (IPR032413) states the same. So in a way they are related in structure, which led us to believe that there is no mistake in the unit definition and this is the reason why we obtained a mixed cluster. We decided to increase the TM-score to a value higher or equal to 0.86 and we obtained three different sub clusters shown in Figure 79, Figure 80 and Figure 81.

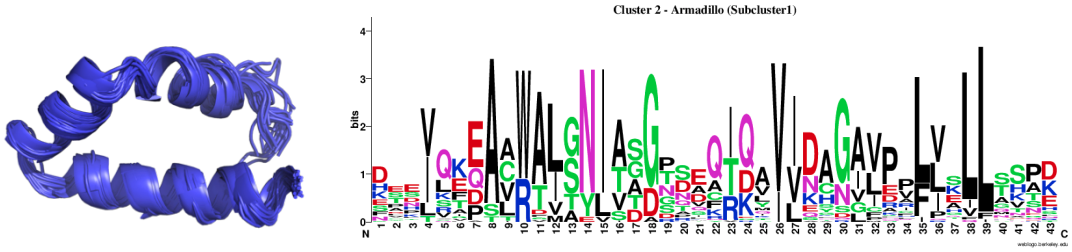


Figure 79 Armadillo sub cluster 1 sequences Logo and structural alignment

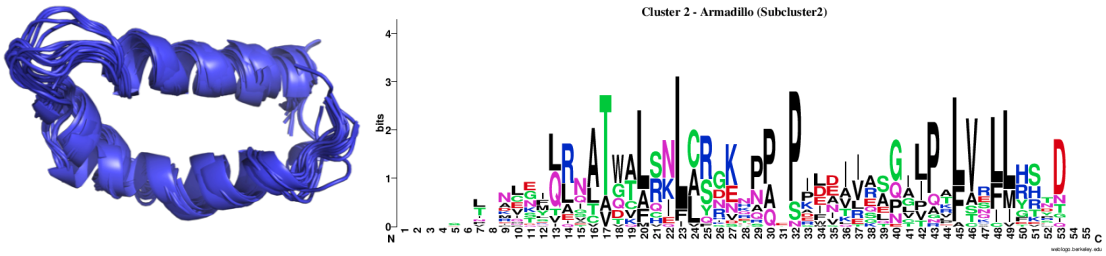


Figure 80 Armadillo sub cluster 2 sequences Logo and structural alignment

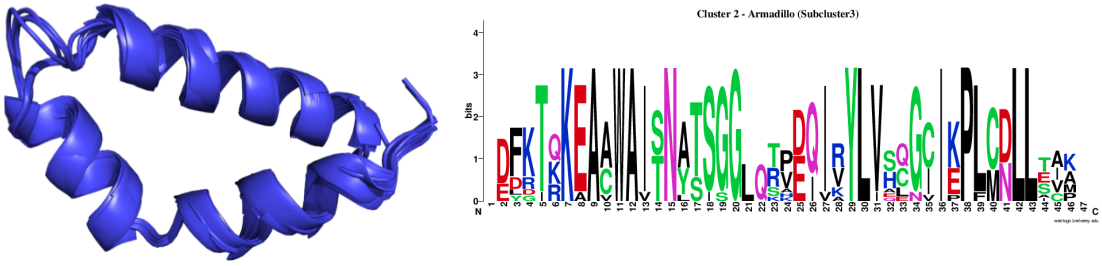


Figure 81 Armadillo sub cluster 3 sequences Logo and structural alignment

This new clusterization, gave us three different unit structures, with different conserved residues and clearly a cleaner sequence alignment. An interesting observation is that all their secondary structures have three α-helices but in each one a different twist is present.

The average length of the unit is 47 residue while in PFAM the average length is 40, we think that this difference if because of the Importin present in our clusters which is the reason why we did try to separate them in a different cluster but we did not succeeded, the motive is that the units of armadillo and importin are pretty similar in sequence and structure based on our observations.

4.3.7. HEAT

Finally the last cluster contains HEAT plus Importin, structural alignment of all units (Figure 82) shows us that the unaligned regions are mostly in the N and C terminal regions. So we decided to use a TM-score value higher of equal than 0.76, and obtain two sub clusters, shown in Figure 83 and Figure 84.

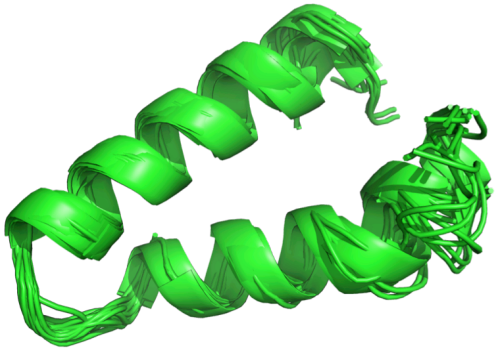


Figure 82 Heat all cluster units structural alignment

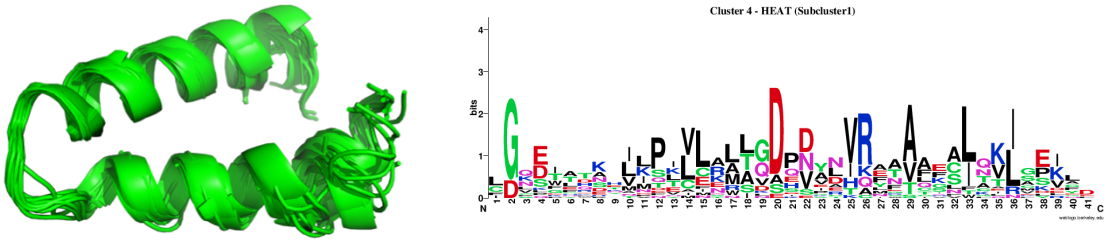


Figure 83 HEAT subc luster 1 sequences Logo and structural alignment

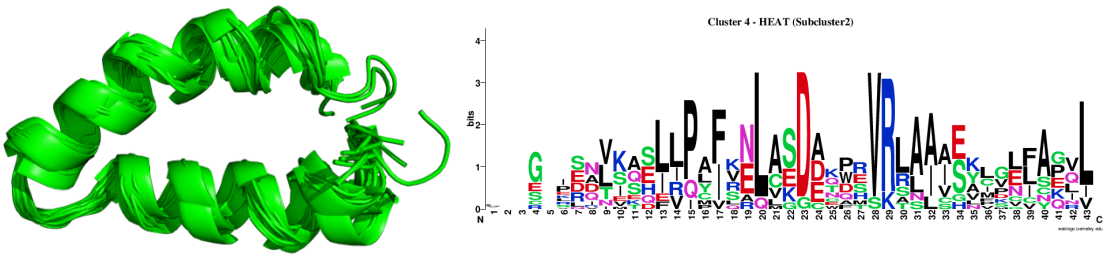


Figure 84 HEAT sub cluster 2 sequences Logo and structural alignment

In the new sub clusters we still have Importing mixed with HEAT units, but different unit structures in each of them. For the HEAT family, PFAM presents four different families, HEAT (PF02985) and HEAT_2(PF13646) as HEAT repeats and HEAT_EZ(PF13513) together with HEAT_PBS(PF03130) as HEAT-like repeat. Interpro (IPR000357) states that: “it is a tandemly repeated of 37-47 amino acid long that occurs in cytoplasmatic proteins, including proteins as huntingtin. Arrays of this ensemble consist of 3 to 36 units forming a rod-like helical structure and appear to function as protein-protein interaction surface. And they can be also involved in intracellular transport processes”. In the sub cluster 1, we can observe two α-helices while in the second sub cluster it seems two have three α-helices yet in reality the second α-helix has a turn. Another interesting observation is that both sub clusters present almost the same conserved residues but in different percentages. Looking at the whole structure it seems that the units are parallel between themselves with just a little angle of separation that seems to be the effect of the interactions.

After analyzing all the clusters separately, we are able to say that a new structural sub classification should be proposed, not only for the α-solenoids but probably also for the rest of the structural subclasses. But in general this analysis is important because we are able to relate lengths (Table 10), residues type and positions with a specific type of repeat unit which could help us a lot not only in the understanding of the protein and its function but also in their design.

Table 10 Statistics for units in each cluster

Ensemble	Units in cluster	Average length	Std dev
Ankyrin	306	32,71	2,00
Armadillo	195	44,13	3,90
HEAT	78	38,23	1,73
TPR	89	33,95	1,71
PPTA	29	35,10	1,78
Pumilio	27	35,30	2,23
SEL1	17	34,41	3,57

Table 11 Seeds, sequences and units found after a HMMER search using our's HMMs and Pfam's

	HMM	**PFAM**
ensemble name	**Ankyrin**	
Seed	438	1554
Uniprot	122160	Unable to retrieve
Units	Unable to retrieve	Unable to retrieve
	HMM	**PFAM**
ensemble name	**Armadillo**	
Seed	245	378
Uniprot	10838	20355
Units	37398	Unable to retrieve
	HMM	**PFAM**
ensemble name	**HEAT**	
Seed	109	937
Uniprot	99732	Unable to retrieve
Units	Unable to retrieve	Unable to retrieve
	HMM	**PFAM**
ensemble name	**TPR**	
Seed	108	3112
Uniprot	61087	Unable to retrieve
Units	41214	Unable to retrieve
	HMM	**PFAM**
ensemble name	**PPTA**	
Seed	48	834
Uniprot	2694	3306
Units	9192	12996
	HMM	**PFAM**
ensemble name	**Pumilio**	
Seed	36	50
Uniprot	5796	6880
Units	30169	34457
	HMM	**PFAM**
ensemble name	**SEL1**	
Seed	17	170
Uniprot	22343	32706
Units	65948	111678

Trying to proof the efficiency of our models, we ran our HMM against UniprotKB and the results show that our models are more efficient than the PFAM ones. Using HMMER search (R. D. Finn, Clements, and Eddy 2011) we found, in most of the cases, more than 70% of Pfam sequences but using much less units as seed (Table 11). This led us to believe that maybe if we increase our dataset to all 1329 α-solenoids in RepeatsDB we probably might be able to find as much sequences as Pfam but using more efficient HMM.

4.4. Manipulating proteins structures with Victor library

(L. Hirsh et al. 2015), Victor library is provided as a GitHub repository with source files and unit test, it also includes extensive online documentation, including a Wiki with help files and tutorials, examples and Doxigen documentation. It is composed of more than 60000 lines of code and is still expanding. All the code is the result of a refactoring process in which all the components of the library maintain the same internal structure. During this refactoring process we include comments for all the methods and a documentation of how they work and communicate. The refactoring process was really complex because we had 15 years of code to evaluate, this is the reason why we also include unit test and examples for the methods. The complexity of Victor is now reduced to different packages; in that way we have encapsulated information that will enable inexperienced users to develop advanced tools. The first package is Biopool, the Biopool class contains methods to parse PDB files. It is the main module of the library.

To demonstrate the range of possible applications Victor provides three main components: Energy, Align and Lobo. Energy contains everything necessary to evaluate protein structures. Using implemented methods it is possible to obtain solvation potential, torsion angles from a PDB and its normalized energy. Included in this section of the library are two published methods FRST (S. C. E. Tosatto 2005), (http://protein.cribi.unipd.it/frst/), that serves to validate energy inside of a protein structure by computing both, an overall and a per-residue

energy profile of a protein structure. The second method is TAP (S. C. Tosatto and Battistutta 2007), that serves to validate local torsion angles of a protein structure calculating both an overall conformational score of a protein structure and a confidence estimate. Both published methods can be used as a guide to develop new methods.

More examples can be found in the Align directory, this package provides basic sequence alignment algorithms (S. C. E. Tosatto et al. 2006). Align is a tool designed for performing sequence alignments in a wide variety of combinations. It implements sequence-to-sequence, sequence to profile and profile to profile alignments with optional support of secondary structure. Different alignment options are freely selectable and include alignment types like local, global, free-shift and number of sub-optimal results to report. The secondary structures can be either provided by the user or automatically performed by the server using PSIPRED. Different profile-profile scoring schemes (Wang and Dunbrack 2004) used in CASP to detect homologous protein sequences are also implemented. In Align we can also find variable gap penalties with additional terms for sequence to structure fit and advance weight scheme such as PSIC (Sunyaev et al. 1999) all default values are optimized by and extensive benchmark.

Finally the Lobo component contains and application of ab initio loop modeling using a fast divide and conquer algorithm (S. C. E. Tosatto et al. 2002) . It is a fast ab-initio method for modeling local segments in protein structures. The algorithm uses a database of recalculated look-up tables, which represent a large set of possible conformations for loop segments of variable length. The target loop is recursively decomposed until the resulting conformations are small enough to be compiled analytically. The algorithm, which is not restricted to any specific loop length, generates a ranked set of loop conformations in 20 - 180 seconds on a desktop PC. The prediction quality is evaluated in terms of global RMSD. Depending on loop length the top prediction varies between 1.06 A RMSD for three-residue loops and 3.72 A RMSD for eight-residue loops. Due to its speed the method may also be useful to generate alternative starting conformations for complex simulations. Using the methods its possible to not only obtain the torsion angles from a PDB, but also to cluster angle data,

generate clusters lookup tables, generate LUTs using Ramachandran clustered data, analyze backbone geometry of a PDB, and of course identify and model loops in a PDB. It could be easily extended for structure prediction in combination with statistical potentials as target function.

By all means, this open source project is devoted to the structural bioinformatics community and provides a unique combination of methods for sequence and structure manipulations and considering all published research we are able to confirm the accuracy of the library and its capability to be extended, this is the reason why, in the future, it will be the main tool for developing a homology modeler of repeat proteins using repeat unit fragments as template.

5. Conclusions and future work

Over the last decade, numerous studies have demonstrated the fundamental importance of tandem repeat proteins (TRP) in many biological processes (Andrade, Perez-Iratxeta, and Ponting 2001). It is known that repeat proteins are a widespread class of non-globular proteins carrying heterogeneous functions involved in several diseases. One of the most frequent problems in study of biology is the functional characterization of a protein and it is usually solved by analyzing the three-dimensional (3D) structure. The experimental determination of the 3D structure is time consuming and technically difficult. For this reason structure prediction by homology modeling offers a fast alternative to experimental approaches. However homology modeling is not feasible for tandem repeats proteins because it is known that these kind of proteins are degenerated and usually it is hard to find a template based on sequence similarity.

For these reasons, this document is focused on algorithms oriented toward repeat unit prediction, presented three different publications (Paladin et al. 2016; Layla Hirsh et al. 2016; L. Hirsh et al. 2015) and one unpublished work that are the first steps of a much bigger picture and future research.
To increase Repeats DB annotations we adopted an innovative approach and created ReUPred (Layla Hirsh et al. 2016), a predictor that identifies repeat units and classifies repeat protein structures. One of the results of this approach is the construction of the Structural Repeat Unit Library that contains different repeat unit fragments representing the structure diversity of known TRPs. This library is continuously updated and optimized, as soon as new structures are deposited in the PDB we execute the predictor and new unit fragments are included in SRUL. This library and the predictor are two products that will be continuously updated and optimized. ReUPred and RepeatsDB provide new information about repeat units and new unclassified repeat protein structures, which will increase our knowledge about repeat proteins. As the quality of SRUL increases predictions of ReUPred will also get better and our information will be more accurate.

The manual curation process should be also done because it is the only way in which new subclasses can be identified. However, it should be considerably faster than before since ReUPred predictions can guide expert curators. Furthermore, as long as new subclasses are identified, the TRPs schema (Figure 50) can be updated as is the case of the box and align prism.

The next step with ReUPred is, as mentioned before, to create a web service for the identification of repeats based on the sequence thanks to the generated HMM. In addition, we want to provide more information of the residue conservations and the positions of structural twist and turns. Moreover, we will provide information about the function by exploiting annotation available from SRUL templates.

ReUPred is based in the Victor C++ (L. Hirsh et al. 2015) library that allows us to manipulate proteins. The final goal of the library is to be able to provide ab initio modeling of repeat proteins. This is a long-term goal, but this library represents the first step to do so.

Another important result of ReUPred is the large-scale generation of high quality data that allowed us to update RepeatsDB, increasing the amount of annotated TRPs 20 times. RepeatsDB 2.0 (Paladin et al. 2016) has now more than 50% of the entries with unit definition manually validated by expert curation. Our final goal is to have 100% of RepeatsDB entries manually curated and an automatic pipeline for its continuous update. To further improve RepeatsDB quality we decided to provide a finer classification at the subclass level. We focused on α-solenoids that represent the most abundant fold in repeat proteins. Our hypothesis that inside this subclass repeats unit share same structure was confirmed and we were able to create different hidden Markov models (HMM) for each ensemble (structural cluster). This are based in the unit structures of already known and identify repeat proteins and will give us the possibility of classify proteins using just their sequence. Moreover, it will be possible to include these HMMs in the ReUPred predictor thus reducing the execution time and increasing the quality of the predictions. Furthermore, the SRUL fragments can be used for ab initio modeling by softwares like Rosetta and for protein engineering in general. So far the subclass analysis was limited to α-solenoids but will be extended to the rest of RepeatsDB subclasses. The

structure of the units seems to be defined mainly by specific conserved residues. We hypothesize that this is true for all subclasses and that this information is the key for TPR design.

Finally during the curation process, we identified new structural subclasses, as is the case of “Sandwich beads” (class V) and “αβ-trefoil” (class IV) among others. We suspect that as new structures are deposited in PDB we will be able to create a new level of structural classification. Maybe by using a unit’s secondary structure, or the idea of having a limited possible number of repeat units or a infinitive number, but in any case, this analysis would be also part of a future research.

In conclusion, this document is just the starting point to understand repeat proteins, how they work, how they can be modeled and designed. Without doubt, to understand how they are related to diseases and their function.

Bibliography

Abraham, Anne-Laure, Eduardo P C Rocha, and Joël Pothier. 2008. "Swelfe: A Detector of Internal Repeats in Sequences and Structures." *Bioinformatics* 24 (13): 1536–37. doi:10.1093/bioinformatics/btn234.

Alberts, Bruce, ed. 2002. *Molecular Biology of the Cell*. 4th ed. New York: Garland Science.

Andersen, Niels H. 2001. "Protein Structure, Stability, and Folding. Methods in Molecular Biology. Volume 168 Edited by Kenneth P. Murphy (University of Iowa College of Medicine). Humana Press: Totowa, New Jersey. 2001. Ix + 252 Pp. $89.50. ISBN 0-89603-682-0." *Journal of the American Chemical Society* 123 (51): 12933–34. doi:10.1021/ja0152815.

Andrade, M A, C Perez-Iratxeta, and C P Ponting. 2001. "Protein Repeats: Structures, Functions, and Evolution." *Journal of Structural Biology* 134 (2–3): 117–31. doi:10.1006/jsbi.2001.4392.

Andrade, M A, C Petosa, S I O'Donoghue, C W Müller, and P Bork. 2001. "Comparison of ARM and HEAT Protein Repeats." *Journal of Molecular Biology* 309 (1): 1–18. doi:10.1006/jmbi.2001.4624.

Andrade, M A, C P Ponting, T J Gibson, and P Bork. 2000. "Homology-Based Method for Identification of Protein Repeats Using Statistical Significance Estimates." *Journal of Molecular Biology* 298 (3): 521–37. doi:10.1006/jmbi.2000.3684.

Andreeva, Antonina, Dave Howorth, John-Marc Chandonia, Steven E Brenner, Tim J P Hubbard, Cyrus Chothia, and Alexey G Murzin. 2008. "Data Growth and Its Impact on the SCOP Database: New Developments." *Nucleic Acids Research* 36 (Database issue): D419-425. doi:10.1093/nar/gkm993.

Anfinsen, C. B., John T. Edsall, Frederic M. Richards, and Eddie Morild, eds. 1981. *Null*. Advances in Protein Chemistry; 34. New York: Academic Press.

Apic, Gordana, Julian Gough, and Sarah A Teichmann. 2001. "Domain Combinations in Archaeal, Eubacterial and Eukaryotic Proteomes." *Journal of Molecular Biology* 310 (2): 311–25. doi:10.1006/jmbi.2001.4776.

Bairoch, A. 2004. "The Universal Protein Resource (UniProt)." *Nucleic Acids Research* 33 (Database issue): D154–59. doi:10.1093/nar/gki070.

Berman, Helen M., John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. 2000. "The Protein Data Bank." *Nucleic Acids Research* 28 (1): 235–42. doi:10.1093/nar/28.1.235.

Björklund, Åsa K., Diana Ekman, and Arne Elofsson. 2006. "Expansion of Protein Domain Repeats." *PLoS Computational Biology* 2 (8): 0959–70. doi:10.1371/journal.pcbi.0020114.

Björklund, Asa K., Diana Ekman, and Arne Elofsson. 2006. "Expansion of Protein Domain Repeats." *PLoS Computational Biology* 2 (8): e114. doi:10.1371/journal.pcbi.0020114.

Brunette, T. J., Fabio Parmeggiani, Po-Ssu Huang, Gira Bhabha, Damian C. Ekiert, Susan E. Tsutakawa, Greg L. Hura, John A. Tainer, and David Baker. 2015. "Exploring the Repeat Protein Universe through Computational Protein Design." *Nature* 528 (7583): 580–84. doi:10.1038/nature16162.

Chevanne, Damien, Sven J Saupe, Corinne Clavé, and Mathieu Paoletti. 2010. "WD-Repeat Instability and Diversification of the Podospora Anserina Hnwd Non-Self Recognition Gene Family." *BMC Evolutionary Biology* 10 (1): 134. doi:10.1186/1471-2148-10-134.

Crooks, G. E. 2004. "WebLogo: A Sequence Logo Generator." *Genome Research* 14

(6): 1188–90. doi:10.1101/gr.849004.
Cuff, Alison L, Ian Sillitoe, Tony Lewis, Andrew B Clegg, Robert Rentzsch, Nicholas Furnham, Marialuisa Pellegrini-Calace, David Jones, Janet Thornton, and Christine A Orengo. 2011. "Extending CATH: Increasing Coverage of the Protein Structure Universe and Linking Structure with Function." *Nucleic Acids Research* 39 (Database issue): D420-426. doi:10.1093/nar/gkq1001.
Di Domenico, Tomás, Emilio Potenza, Ian Walsh, R. Gonzalo Parra, Manuel Giollo, Giovanni Minervini, Damiano Piovesan, et al. 2014a. "RepeatsDB: A Database of Tandem Repeat Protein Structures." *Nucleic Acids Research* 42 (Database issue): D352-357. doi:10.1093/nar/gkt1175.
Domenico, Tomás Di, Emilio Potenza, Ian Walsh, R. Gonzalo Parra, Manuel Giollo, Giovanni Minervini, Damiano Piovesan, et al. 2014b. "RepeatsDB: A Database of Tandem Repeat Protein Structures." *Nucleic Acids Research* 42 (Database issue): D352-357. doi:10.1093/nar/gkt1175.
Espada, R., R. G. Parra, M. J. Sippl, T. Mora, A. M. Walczak, and D. U. Ferreiro. 2015. "Repeat Proteins Challenge the Concept of Structural Domains." *Biochemical Society Transactions* 43 (5): 844–49. doi:10.1042/BST20150083.
Finn, R. D., J. Clements, and S. R. Eddy. 2011. "HMMER Web Server: Interactive Sequence Similarity Searching." *Nucleic Acids Research* 39 (suppl): W29–37. doi:10.1093/nar/gkr367.
Finn, Robert D., Teresa K. Attwood, Patricia C. Babbitt, Alex Bateman, Peer Bork, Alan J. Bridge, Hsin-Yu Chang, et al. 2017. "InterPro in 2017-beyond Protein Family and Domain Annotations." *Nucleic Acids Research* 45 (D1): D190–99. doi:10.1093/nar/gkw1107.
Forrer, Patrik, H. Kaspar Binz, Michael T. Stumpp, and Andreas Plückthun. 2004. "Consensus Design of Repeat Proteins." *ChemBioChem* 5 (2): 183–89. doi:10.1002/cbic.200300762.
Fournier, David, Gareth A. Palidwor, Sergey Shcherbinin, Angelika Szengel, Martin H. Schaefer, Carol Perez-Iratxeta, and Miguel A. Andrade-Navarro. 2013. "Functional and Genomic Analyses of Alpha-Solenoid Proteins." *PLoS ONE* 8 (11): e79894. doi:10.1371/journal.pone.0079894.
Fu, Limin, Beifang Niu, Zhengwei Zhu, Sitao Wu, and Weizhong Li. 2012. "CD-HIT: Accelerated for Clustering the next-Generation Sequencing Data." *Bioinformatics (Oxford, England)* 28 (23): 3150–52. doi:10.1093/bioinformatics/bts565.
Gamma, Erich, ed. 1995. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley Professional Computing Series. Reading, Mass: Addison-Wesley.
Gemayel, Rita, Marcelo D. Vinces, Matthieu Legendre, and Kevin J. Verstrepen. 2010. "Variable Tandem Repeats Accelerate Evolution of Coding and Regulatory Sequences." *Annual Review of Genetics* 44 (1): 445–77. doi:10.1146/annurev-genet-072610-155046.
George, Richard A., and Jaap Heringa. 2000. "The REPRO Server: Finding Protein Internal Sequence Repeats through the Web." *Trends in Biochemical Sciences* 25 (10): 515–17. doi:10.1016/S0968-0004(00)01643-1.
Grant, C. E., T. L. Bailey, and W. S. Noble. 2011. "FIMO: Scanning for Occurrences of a given Motif." *Bioinformatics* 27 (7): 1017–18. doi:10.1093/bioinformatics/btr064.
Grove, Tijana Z, Aitziber L Cortajarena, and Lynne Regan. 2008. "Ligand Binding by Repeat Proteins: Natural and Designed." *Current Opinion in Structural Biology* 18

(4): 507–15. doi:10.1016/j.sbi.2008.05.008.
Heger, A, and L Holm. 2000. “Rapid Automatic Detection and Alignment of Repeats in Protein Sequences.” *Proteins* 41 (2): 224–37.
Higgs, Paul G., and Teresa K. Attwood. 2005. *Bioinformatics and Molecular Evolution*. Malden, MA: Blackwell Pub.
Hirsh, L., D. Piovesan, M. Giollo, C. Ferrari, and S. C. E. Tosatto. 2015. “The Victor C++ Library for Protein Representation and Advanced Manipulation.” *Bioinformatics* 31 (7): 1138–40. doi:10.1093/bioinformatics/btu773.
Hirsh, Layla, Damiano Piovesan, Lisanna Paladin, and Silvio C. E. Tosatto. 2016. “Identification of Repetitive Units in Protein Structures with ReUPred.” *Amino Acids* 48 (6): 1391–1400. doi:10.1007/s00726-016-2187-2.
Höcker, Birte. 2014. “Design of Proteins from Smaller Fragments — Learning from Evolution.” *Current Opinion in Structural Biology*, Membranes / Engineering and design, 27 (August): 56–62. doi:10.1016/j.sbi.2014.04.007.
Hrabe, Thomas, and Adam Godzik. 2014. “ConSole: Using Modularity of Contact Maps to Locate Solenoid Domains in Protein Structures.” *BMC Bioinformatics* 15: 119. doi:10.1186/1471-2105-15-119.
Javadi, Yalda, and Laura S Itzhaki. 2013. “Tandem-Repeat Proteins: Regularity plus Modularity Equals Design-Ability.” *Current Opinion in Structural Biology* 23 (4): 622–31. doi:10.1016/j.sbi.2013.06.011.
Jorda, J., and A. V. Kajava. 2009. “T-REKS: Identification of Tandem REpeats in Sequences with a K-meanS Based Algorithm.” *Bioinformatics* 25 (20): 2632–38. doi:10.1093/bioinformatics/btp482.
Jorda, Julien, Bin Xue, Vladimir N Uversky, and Andrey V Kajava. 2010. “Protein Tandem Repeats - the More Perfect, the Less Structured.” *The FEBS Journal* 277 (12): 2673–82. doi:10.1111/j.1742-464X.2010.07684.x.
Kajava, A. V. 2001. “Review: Proteins with Repeated Sequence--Structural Prediction and Modeling.” *Journal of Structural Biology* 134 (2–3): 132–44. doi:10.1006/jsbi.2000.4328.
Kajava, Andrey V. 2012. “Tandem Repeats in Proteins: From Sequence to Structure.” *Journal of Structural Biology* 179 (3): 279–88. doi:10.1016/j.jsb.2011.08.009.
Kajava, Andrey V. 2012. “Tandem Repeats in Proteins: From Sequence to Structure.” *Journal of Structural Biology*, Structural Bioinformatics, 179 (3): 279–88. doi:10.1016/j.jsb.2011.08.009.
Katti, Mukund V., R. Sami-Subbu, Prabhakar K. Ranjekar, and Vidya S. Gupta. 2000. “Amino Acid Repeat Patterns in Protein Sequences: Their Diversity and Structural-Functional Implications.” *Protein Science* 9 (6): 1203–9. doi:10.1110/ps.9.6.1203.
Kobe, B., and A. V. Kajava. 2000. “When Protein Folding Is Simplified to Protein Coiling: The Continuum of Solenoid Protein Structures.” *Trends in Biochemical Sciences* 25 (10): 509–15.
Kobe, Bostjan, and Andrey V. Kajava. 2000. “When Protein Folding Is Simplified to Protein Coiling: The Continuum of Solenoid Protein Structures.” *Trends in Biochemical Sciences* 25 (10): 509–15. doi:10.1016/S0968-0004(00)01667-4.
Konagurthu, Arun S, James C Whisstock, Peter J Stuckey, and Arthur M Lesk. 2006. “MUSTANG: A Multiple Structural Alignment Algorithm.” *Proteins* 64 (3): 559–74. doi:10.1002/prot.20921.
Lee, Robin van der, Marija Buljan, Benjamin Lang, Robert J. Weatheritt, Gary W. Daughdrill, A. Keith Dunker, Monika Fuxreiter, et al. 2014. “Classification of

Intrinsically Disordered Regions and Proteins." *Chemical Reviews* 114 (13): 6589–6631. doi:10.1021/cr400525m.
Liao, Daiqing. 1999. "Concerted Evolution: Molecular Mechanism and Biological Implications." *The American Journal of Human Genetics* 64 (1): 24–30. doi:10.1086/302221.
Marco Milán et al. n.d. "JAK/STAT Controls Organ Size and Fate Specification by Regulating Morphogen Production and Signalling." *Nature Communications*. http://www.nature.com/articles/ncomms13815.
Marcotte, E M, M Pellegrini, T O Yeates, and D Eisenberg. 1999. "A Census of Protein Repeats." *Journal of Molecular Biology* 293 (1): 151–60. doi:10.1006/jmbi.1999.3136.
Marcotte, Edward M., Matteo Pellegrini, Ho-Leung Ng, Danny W. Rice, Todd O. Yeates, and David Eisenberg. 1999. "Detecting Protein Function and Protein-Protein Interactions from Genome Sequences." *Science* 285 (5428): 751–53. doi:10.1126/science.285.5428.751.
Metalloproteins: Structural Aspects. 1991. Advances in Protein Chemistry 42. San Diego: Acad. Press.
Murray, Kevin B, Denise Gorse, and Janet M Thornton. 2002. "Wavelet Transforms for the Characterization and Detection of Repeating Motifs." *Journal of Molecular Biology* 316 (2): 341–63. doi:10.1006/jmbi.2001.5332.
Murray, Kevin B, William R Taylor, and Janet M Thornton. 2004. "Toward the Detection and Validation of Repeats in Protein Structure." *Proteins* 57 (2): 365–80. doi:10.1002/prot.20202.
Newman, Aaron M, and James B Cooper. 2007. "XSTREAM: A Practical Algorithm for Identification and Architecture Modeling of Tandem Repeats in Protein Sequences." *BMC Bioinformatics* 8 (1): 382. doi:10.1186/1471-2105-8-382.
Paladin, Lisanna, Layla Hirsh, Damiano Piovesan, Miguel A. Andrade-Navarro, Andrey V. Kajava, and Silvio C.E. Tosatto. 2016. "RepeatsDB 2.0: Improved Annotation, Classification, Search and Visualization of Repeat Protein Structures." *Nucleic Acids Research*, November. doi:10.1093/nar/gkw1136.
Pearl, Frances, Annabel Todd, Ian Sillitoe, Mark Dibley, Oliver Redfern, Tony Lewis, Christopher Bennett, et al. 2005. "The CATH Domain Structure Database and Related Resources Gene3D and DHS Provide Comprehensive Domain Family Information for Genome Analysis." *Nucleic Acids Research* 33 (Database issue): D247-251. doi:10.1093/nar/gki024.
Pellegrini, Marco. 2015. "Tandem Repeats in Proteins: Prediction Algorithms and Biological Role." *Frontiers in Bioengineering and Biotechnology* 3 (September). doi:10.3389/fbioe.2015.00143.
Pellegrini, Marco, Maria Elena Renda, and Alessio Vecchio. 2012. "Ab Initio Detection of Fuzzy Amino Acid Tandem Repeats in Protein Sequences." *BMC Bioinformatics* 13 (Suppl 3): S8. doi:10.1186/1471-2105-13-S3-S8.
Piovesan, Damiano, Giovanni Minervini, and Silvio C. E. Tosatto. 2016. "The RING 2.0 Web Server for High Quality Residue Interaction Networks." *Nucleic Acids Research*, May, gkw315. doi:10.1093/nar/gkw315.
Privileged Scaffolds in Medicinal Chemistry: Design, Synthesis, Evaluation. 2016. RSC Drug Discovery. Cambridge, England: Royal Society of Chemistry.
Raven, Peter H. 2014. *Biology*. Tenth edition. New York, NY: McGraw-Hill.
Sabarinathan, R., Raunak Basu, and K. Sekar. 2010. "ProSTRIP: A Method to Find Similar Structural Repeats in Three-Dimensional Protein Structures."

Computational Biology and Chemistry 34 (2): 126–30. doi:10.1016/j.compbiolchem.2010.03.006.
Schüler, Andreas, and Erich Bornberg-Bauer. 2016. "Evolution of Protein Domain Repeats in Metazoa." *Molecular Biology and Evolution* 33 (12): 3170–82. doi:10.1093/molbev/msw194.
Soding, J., M. Remmert, and A. Biegert. 2006. "HHrep: De Novo Protein Repeat Detection and the Origin of TIM Barrels." *Nucleic Acids Research* 34 (Web Server): W137–42. doi:10.1093/nar/gkl130.
Sunyaev, S. R., F. Eisenhaber, I. V. Rodchenkov, B. Eisenhaber, V. G. Tumanyan, and E. N. Kuznetsov. 1999. "PSIC: Profile Extraction from Sequence Alignments with Position-Specific Counts of Independent Observations." *Protein Engineering* 12 (5): 387–94.
Szklarczyk, Radek, and Jaap Heringa. 2004. "Tracking Repeats Using Significance and Transitivity." *Bioinformatics* 20 Suppl 1 (August): i311-317. doi:10.1093/bioinformatics/bth911.
Tompa, Peter. 2002. "Intrinsically Unstructured Proteins." *Trends in Biochemical Sciences* 27 (10): 527–33.
Tompa, Peter, Monika Fuxreiter, Christopher J Oldfield, Istvan Simon, A Keith Dunker, and Vladimir N Uversky. 2009. "Close Encounters of the Third Kind: Disordered Domains and the Interactions of Proteins." *BioEssays: News and Reviews in Molecular, Cellular and Developmental Biology* 31 (3): 328–35. doi:10.1002/bies.200800151.
Tosatto, Silvio C E. 2005. "The Victor/FRST Function for Model Quality Estimation." *Journal of Computational Biology* 12 (10): 1316–27. doi:10.1089/cmb.2005.12.1316.
Tosatto, Silvio C. E., Alessandro Albiero, Alessandra Mantovan, Carlo Ferrari, Eckart Bindewald, and Stefano Toppo. 2006. "Align: A C++ Class Library and Web Server for Rapid Sequence Alignment Prototyping." *Current Drug Discovery Technologies* 3 (3): 167–73.
Tosatto, Silvio C E, Eckart Bindewald, Jürgen Hesser, and Reinhard Männer. 2002. "A Divide and Conquer Approach to Fast Loop Modeling." *Protein Engineering* 15 (4): 279–86.
Tosatto, Silvio CE, and Roberto Battistutta. 2007. "TAP Score: Torsion Angle Propensity Normalization Applied to Local Protein Structure Evaluation." *BMC Bioinformatics* 8 (May): 155. doi:10.1186/1471-2105-8-155.
Touw, Wouter G., Coos Baakman, Jon Black, Tim A. H. te Beek, E. Krieger, Robbie P. Joosten, and Gert Vriend. 2015. "A Series of PDB-Related Databanks for Everyday Needs." *Nucleic Acids Research* 43 (D1): D364–68. doi:10.1093/nar/gku1028.
Verstrepen, Kevin J, An Jansen, Fran Lewitter, and Gerald R Fink. 2005. "Intragenic Tandem Repeats Generate Functional Variability." *Nature Genetics* 37 (9): 986–90. doi:10.1038/ng1618.
Viet, Phuong Do, Daniel B. Roche, and Andrey V. Kajava. 2015. "TAPO: A Combined Method for the Identification of Tandem Repeats in Protein Structures." *FEBS Letters* 589 (19 Pt A): 2611–19. doi:10.1016/j.febslet.2015.08.025.
Walsh, Ian, Francesco G. Sirocco, Giovanni Minervini, Tomás Di Domenico, Carlo Ferrari, and Silvio C. E. Tosatto. 2012. "RAPHAEL: Recognition, Periodicity and Insertion Assignment of Solenoid Protein Structures." *Bioinformatics* 28 (24): 3257–64. doi:10.1093/bioinformatics/bts550.
Wang, Guoli, and Roland L. Dunbrack. 2004. "Scoring Profile-to-Profile Sequence

Alignments." *Protein Science* 13 (6): 1612–26. doi:10.1110/ps.03601504.
Wit, Joris de, Weizhe Hong, Liqun Luo, and Anirvan Ghosh. 2011. "Role of Leucine-Rich Repeat Proteins in the Development and Function of Neural Circuits." *Annual Review of Cell and Developmental Biology* 27: 697–729. doi:10.1146/annurev-cellbio-092910-154111.
Wright, Caroline F., Sarah A. Teichmann, Jane Clarke, and Christopher M. Dobson. 2005. "The Importance of Sequence Diversity in the Aggregation and Evolution of Proteins." *Nature* 438 (7069): 878–81. doi:10.1038/nature04195.
Zhang, Y., and J. Skolnick. 2005. "TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score." *Nucleic Acids Research* 33 (7): 2302.
Zhang Y, and Skolnick J. 2005. "TM-Align: A Protein Structure Alignment Algorithm Based on the TM-Score." *Nucleic Acids Res* 33 (7): 2302–2309. doi:10.1093/nar/gki524.

6. Publications

6.1. RepeatsDB 2.0: improved annotation, classification, search and visualization of repeat protein structures

Authors: Lisanna Paladin, Layla Hirsh, Damiano Piovesan, Miguel A. Andrade-Navarro, Andrey V. Kajava and Silvio C.E. Tosatto.
Journal: Nucl. Acids Res. (2016) doi: 10.1093/nar/gkw1136

6.1.1. Abstract

RepeatsDB 2.0 (URL: http://repeatsdb.bio.unipd.it/) is an update of the database of annotated tandem repeat protein structures previously featured in the NAR Database Issue. Repeat proteins are a widespread class of non-globular proteins carrying heterogeneous functions involved in several diseases. Here we provide a new version of RepeatsDB with an improved classification schema including high quality annotations for about 5,400 protein structures. RepeatsDB 2.0 features information on start and end positions for the repeat regions and units for all entries. The extensive growth of repeat unit characterization was possible exploiting the novel ReUPred annotation method over the entire Protein Data Bank. The quality of the data is guaranteed by an extensive manual validation for more than 60% of the entries. The updated web interface includes a new search engine for complex queries, a new entry page for a better overview of structural data. It is possible to compare unit positions, together with secondary structure, fold information and Pfam domains. Moreover, a new classification level has been added on top of the existing classification scheme as an independent layer for sequence similarity relationships at 40%, 60% and 90% identity.

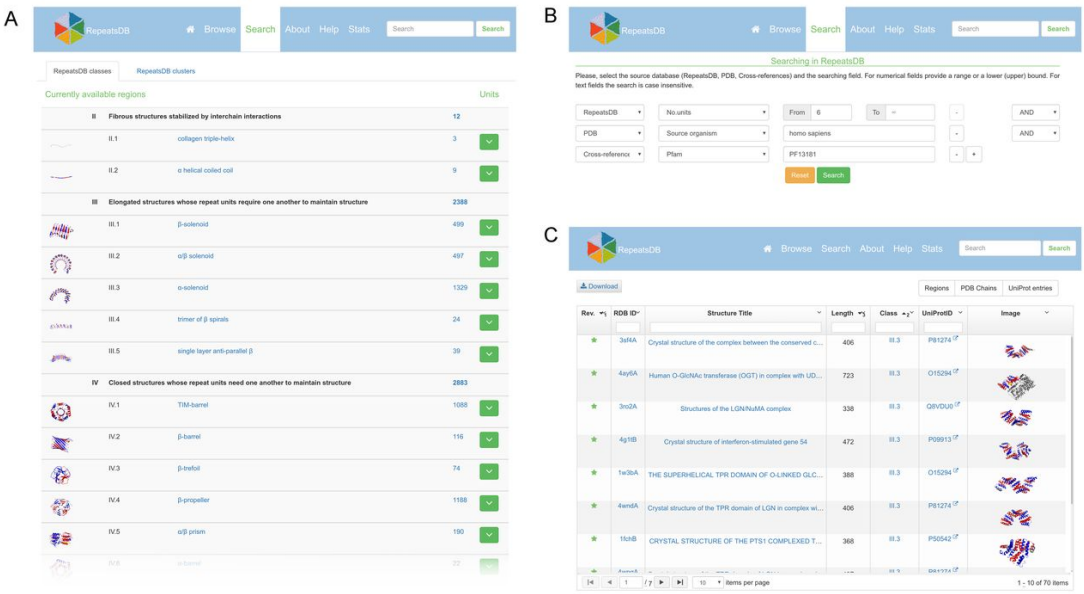
6.1.2. Introduction

Tandem repeat regions in proteins are characterized by a repeated sequence that codes for a modular architecture, where structural

modules are called “units”. Proteins with tandem repeats play important functional roles (1), are abundant in nature and related to major health threats (2–6). Detecting and annotating them appropriately may increase our understanding of mechanisms of pathogenicity (e.g. virulence factors (7)), allow the design of scaffold proteins for engineered ligand binding with multiple applications (e.g. cancer therapy (8)) and generally expand our knowledge of the function and structure of many proteins (e.g. the mineralocorticoid receptor (9)). It is widely accepted that domain structural and functional complexity evolved through fusion, recombination, accretion and repetition of a very small set of elementary functions (10, 11). Therefore, units in tandem repeat proteins represent a fundamental source of information to explain contemporary structural diversity and the physic-chemical properties of highly designable folds (12). However, the identification of the periodicity at the sequence level is an extremely hard task, since repetitive proteins evolve quickly, due to two main reasons. The first is the error-prone process of duplication that originates new repeats , and the second is the intrinsic tendency of flanking identical units to diverge (13). A number of structure-based methods for the identification of repeats has been developed to fill this gap (14–17). RepeatsDB (18), was proposed in 2014 as a database of repeat protein structures and as a resource for high-quality repeat structure annotation. The data was collected with RAPHAEL (19), a state-of-the-art method for the detection of Protein Data Bank (PDB) (20) structures containing repeat regions. The entries were classified into repeat structural classes (21) and further divided into subclasses. The five repeat classes are mainly distinguished by repeat unit length and general structural arrangement, and the subclasses by the secondary structure assignment of the repeat unit. The shortest one or two residue-long repeats, form crystallites and are typically harmful or non-functional in natural organisms. No example of their structure is deposited in the PDB and consequently in RepeatsDB. Class II structures are fibrous proteins with very short units stabilized by interchain interactions, typically collagens and α-helical coiled coils. This second subclass presents various arrangements described in (22).

Class III contains the most typical examples of repeats, elongated structures where repetitive units require one another to maintain structure. The most numerous subclasses in class III are β-, α/β- and α-solenoids. Class IV includes all closed repeat structures. Widespread across all types of organisms, this class includes the TIM-barrel and β-propeller subclasses. Both class III and IV contain units with a length between 10 and 50 residues. The last class V, with unit length > 40 residues, groups “beads on a string” repeats, whose repeat units are large enough to fold independently.

All repeat subclasses are characterized by a strong structural conservation in repeat units frequently not clearly reflected in sequence. This is the reason why domain sequence databases such as Pfam (23) and SMART (24) fail to detect a large number of repeats (25), as most of the largest clusters of human sequence regions not covered by Pfam were found to be repeated (25, 26). RepeatsDB was developed to fill this gap and provides the community with a high-quality resource of reliable datasets of repeat structures for various purposes. The first and most obvious goal that was achieved was to compare the structural classification of repeat with the sequence-based one (26). Other uses of RepeatsDB are the extraction of repeat datasets to discuss specific features (27, 28), the testing of both sequence- and structure-based repeat detection methods and the discussion of the role of proteins with repeats (17, 29–33). The high-quality manually annotated set of RepeatsDB units, the Structural Repeat Unit Library (SRUL), was exploited Repeat Unit Predictor (ReUPred) (34), an algorithm to predict both unit position and classification of repeat regions in a group of entries predicted as repeated. By running ReUPred on the output of RAPHAEL (19), an algorithm designed to detect repeats in protein structures, a number of new entries were identified, classified and annotated with unit positions. RepeatsDB 2.0 includes new annotations, an improved classification and completely redesigned web server and interface, to guarantee the intuitive availability of data and a better user experience in terms of database usability and look-and-feel.



*Retrieving RepeatsDB data. RepeatsDB data can be retrieved in three different ways. (**A**) The 'Browse' page provides the entry point for both the structural hierarchy and sequence clusters. (**B**) The 'Search' page allows the user to perform advanced queries against a range of RepeatsDB-specific and third-party search fields. The input can be simple text or numeric (single value or range) according to the field type and multiple queries can be combined by boolean operators (AND, OR, NOT). Both the 'Browse' and 'Search' pages redirect to the results page (**C**). This page provides a table with the list of retrieved entries and can be further filtered (and sorted) through column header fields. Results can be displayed by PDB chain (default), region or UniProt.*

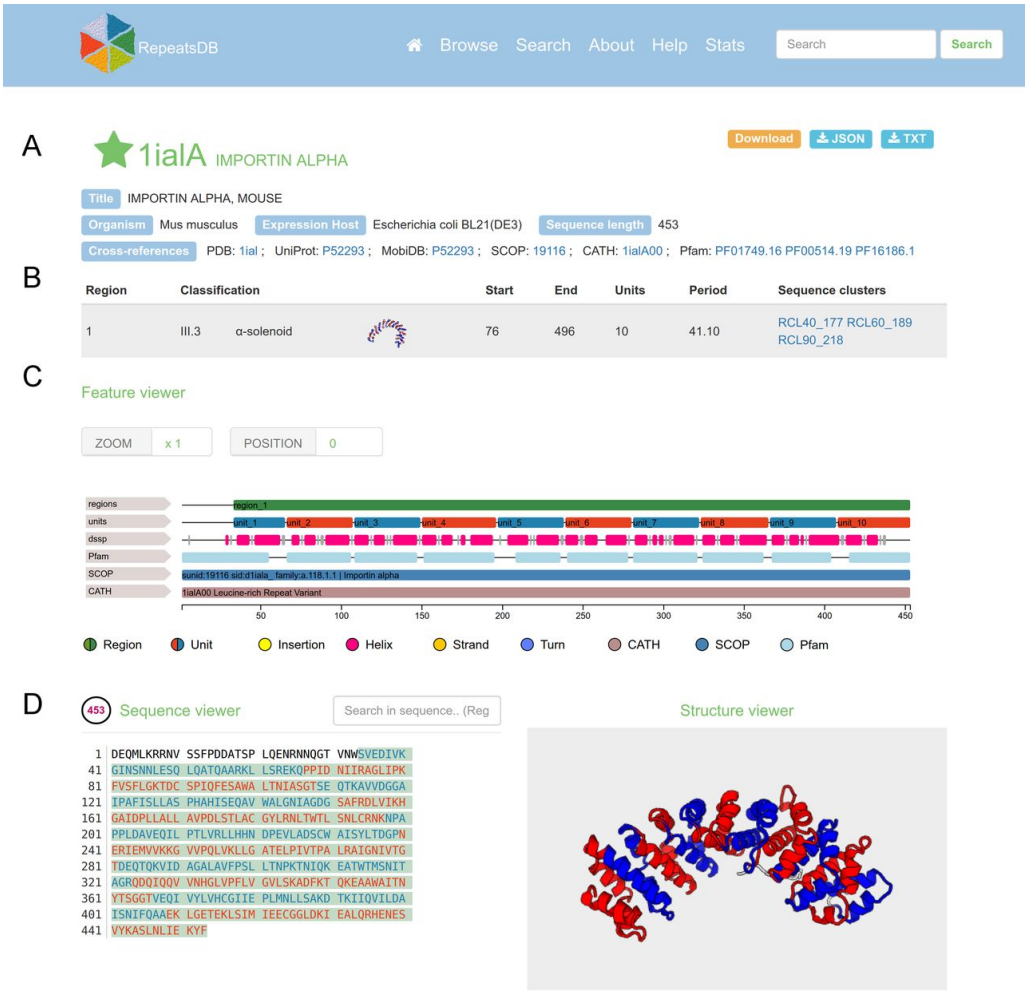
6.1.3. Database description

RepeatsDB 2.0 data have been completely regenerated taking advantage of the new ReUPred predictor (34) for automatic detection of tandem repeat units. In the new database version, all entries are annotated at the unit level, i.e. providing start end position for each repeated segment, and classified at the subclass level. Compared to the old version, unit annotation have grown by more than an order of magnitude. A detailed description of the RepeatsDB annotation pipeline follows.

6.1.4. Data curation

The initial dataset for RepeatsDB is the entire PDB (20). Repeat candidates are extracted with RAPHAEL (19) and processed with

ReUPred (34) to confirm the presence of repeat regions and provide detailed unit information. ReUPred is a predictor able to identify the position of repeated fragments by performing iterative structural alignments against a manually refined library of representative units. ReUPred is also able to assign the class and subclass by transferring this information from the unit library.



*Screenshot of RepeatsDB sample entry page for PDB code 1ialA. The top part of the page (**A**) reports structure information from the PDB and cross-references to third-party databases including UniProt, MobiDB, SCOP, CATH and Pfam (when available). RepeatsDB annotations are available for download both in text and JSON formats on the top-right corner. (**B**) A table provides region details such as structural classification, start/end position, number of units, repeat period and cluster families. (**C**) The feature viewer summarizes available annotation for the PDB reference sequence, i.e. the SEQRES field in the PDB file. An overview of RepeatsDB information (regions, units and insertions) along with secondary structure (DSSP), Pfam, SCOP and CATH tracks (when available) are shown. (**D**) A detailed view of RepeatsDB annotations is highlighted in the sequence and PDB viewers.*

The final dataset available in RepeatsDB 2.0 is the result of an iterative process where the ReUPred library has been refined manually multiple times to resolve conflicts, to improve its ability to generalize and to include newly discovered subclasses. At the end of the process an extensive validation and refinement of the predictions has been carried out by expert visual inspection. More than 60% of the entries has been reviewed and 5 new subclasses created, three for class IV (closed structure) and two for class V (beads on a string).

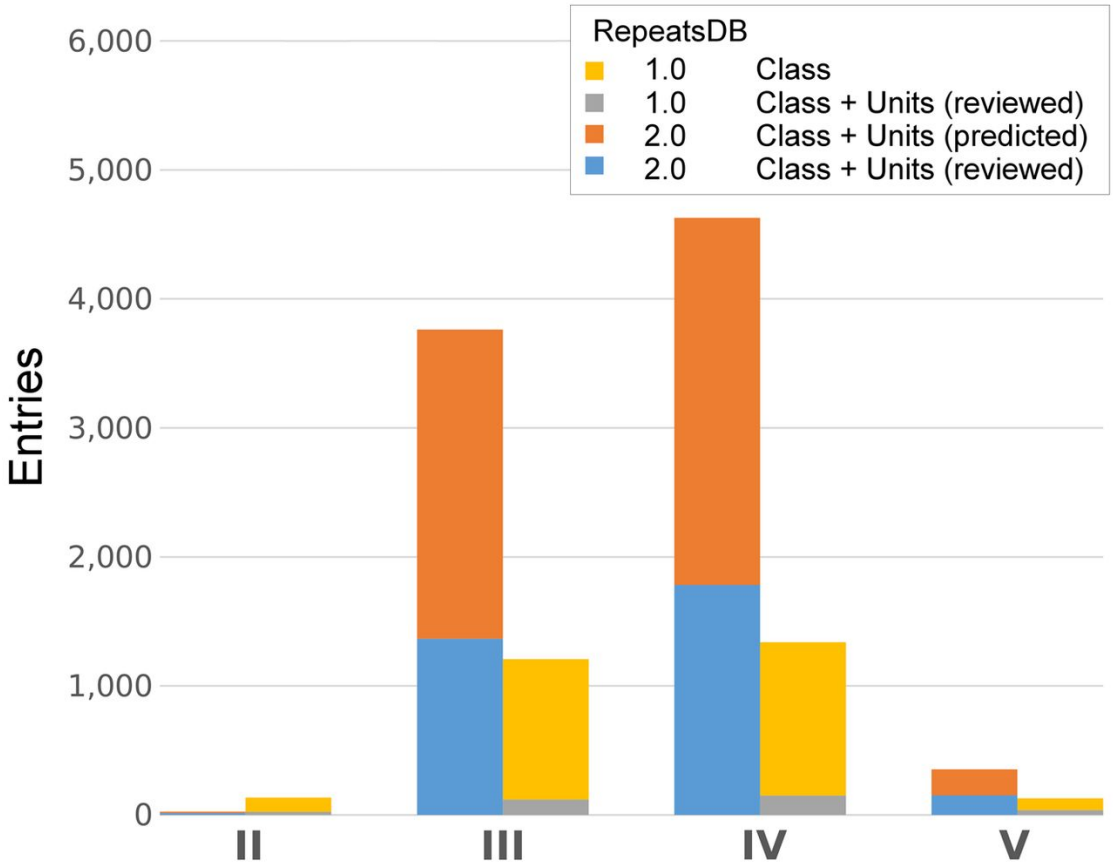
6.1.5. Implementation

RepeatsDB was designed as a multi-tier architecture, with three modules managing data storage, processing and presentation, respectively. Data are stored in a MongoDB database, and processed with Node.js. The server is accessible through a web interface or programmatically exploiting a RESTful architecture. The web interface is designed using Angular.js and Bootstrap frameworks. Dynamic and interactive elements of the entry page are developed using PV for the structure visualization and Bio.js for the sequence features viewer, respectively. Both the database structure and the Node.js server have been completely rewritten to improve efficiency and data reliability. Moreover, all data derived from third party resources have been processed and stored locally to prevent broken dependencies.

6.1.6. Innovations

Apart from the new annotation pipeline, several bugs have been fixed and many improvements have been introduced since the last RepeatsDB release. All positional annotations are now based on SIFTS (35) making them consistent with both PDB (20) and UniProt (36) references. The search engine has been completely redesigned. An intuitive interface allows to perform complex queries using logical operators and guides the user through all possible searching fields. A

new classification level has been added to include evolutionary relationships among different repeat regions. An all-vs.-all alignment of the repeat regions allowed to group them according to sequence similarity and to identify different repeat families. The new classification has been implemented as an independent layer on top of the existing structural features, and it is available at three different identity thresholds (40, 60 and 90%). The web interface allows to navigate entry clusters and so providing an overview of the sequence representativeness inside each structural subclass.



RepeatsDB growth. RepeatsDB 2.0 is compared to the previous release. Entries have unit and subclass annotation, with more than 60% manually reviewed (blue). For the old version, only a tiny fraction of entries have unit definition (cyan) and the rest is mostly annotated only at the class level (yellow).

6.1.7. Database usage

The user interface presents an intuitive summary table providing direct access to all entries by structural class directly from the home page. For a finer search, the user can visit either the “Browse” page that provides subclass access or the “Search” page for generating complex queries (Figure 1, panel A and B). All entry points redirect to the same result page listing the retrieved proteins in a table (Figure 1, panel C). The table can be further filtered by providing additional matching strings in the column headers. The “Browse” page provides also direct access to sequence clusters, where entries are grouped by sequence similarity. The entry page (Figure 2) is much more informative compared to the previous RepeatsDB version, including several cross-links to third party resources. It also integrates several structural features useful for comparing CATH, SCOP, Pfam and DSSP annotations with RepeatsDB data. Regions, units and insertions are provided for all entries and are correctly mapped both to UniProt and PDB reference (SEQRES field in the PDB file) sequences thanks to the SIFTS service. A correct mapping strongly improves RepeatsDB impact since it is now very easy to link repeat data with other sequence features like mutations or post-translational modifications. Thanks to a RESTful architecture, all RepeatsDB data are accessible from external APIs and third party resources through HTTP URLs. Please refer to the ‘Help’ section of the website for details on using the RepeatsDB web services. Customized datasets can be downloaded in JSON or text format using the browse function or RESTful web services.

6.1.8. Statistics

RepeatsDB provides high quality annotation for about 5,400 entries. Figure 3 compares the current RepeatsDB content to the previous version. The chart shows the total number of entries belonging to each class. However, the new version provides unit definition and subclass classification for all entries whereas the old version only for a tiny fraction, 327 entries (cyan bar). Moreover, in RepeatsDB 2.0 more than 60% of the entries has been manually reviewed by expert curators (blue

segment). Further details as the number of regions, units and genes are available in the “Stats” page of the web site.

6.1.9. Conclusion and future work

RepeatsDB was presented in 2014 with the goal to provide the community with a central resource for high-quality tandem repeat protein structure annotation. RepeatsDB has been cited in a number of different studies regarding repeat proteins, and has been used to extract databases for repeat proteins analysis and to test algorithms for repeat proteins annotation. The detailed annotation of entries performed by RepeatsDB curators led to the building of a high quality Structure Repeat Unit Library (SRUL). This library was exploited by the ReUPred algorithm (34) as a gold standard to define unit position in new entries.

The new release of RepeatsDB includes a new annotation pipeline, combining the RAPHAEL algorithm for repeat detection (19) and ReUPred for annotation (34), producing extensive annotation for all entries. The pipeline is fully automated and allows the easy regular update of the database. The iterative execution of the pipeline already demonstrated its efficacy both because it identified a large number of new entries, and because new subclasses were identified and added to the structural classification scheme. RepeatsDB will benefit from regular updates, which will steadily increase the number of available annotations.

6.2. Identification of repetitive units in protein structures with ReUPred

Authors: Layla Hirsh, Damiano Piovesan, Lisanna Paladin, and Silvio C.E. Tosatto

Journal: Amino Acids, June 2016, Volume 48, Issue 6, pp 1391–1400. DOI: 10.1007/s00726-016-2187-2

6.2.1. Abstract

Motivation: Over the last decade, numerous studies have demonstrated the fundamental importance of repeat proteins in many biological processes. A plethora of new repeat structures have also been solved. However, a detailed structural characterization of repetitive elements is completely missing. A first attempt to remedy this has been recently provided by RepeatsDB, but since repeat unit annotation is generated through time consuming manual curation it covers only 3% of the bona fide repeat proteins.

Results: The Repeat Protein Unit Predictor (ReUPred) for the fast automatic prediction of repeat units and the repeat classification is an algorithm exploiting an extensive Structure Repeat Unit Library (SRUL) derived from RepeatsDB. ReUPred finds repetitive elements through an iterative structural search against the SRUL using a divide and conquer strategy. Taking solenoid proteins as a test case, ReUPred outperforms the state of the art for prediction of the unit position, with an accuracy increase of about 9%. It is the first predictor for the finer classification of solenoid structures.

Availability: ReUPred is implemented in Python and supported on Linux. The source code is distributed under the GPL license and freely available from URL: http://protein.bio.unipd.it/reupred/

6.2.2. Introduction

Tandem repeat (TR) proteins are characterized by a repetitive 3D structure successfully exploited by nature in a myriad different cellular pathways and organisms (E M Marcotte et al. 1999). They are widely distributed in archeal, bacterial and eukaryotic proteomes and prevalent in complex organisms. An association was suggested between TR spread and the evolution of multicellularity (Edward M. Marcotte et al. 1999)(Edward M. Marcotte et al. 1999). Characterized by repetitions in

their coding sequence, they are believed to have arisen from the duplication of short coding DNA segments (Andrade et al. 2001). These repetitions in sequence account for a peculiar modular fold architecture (Andrey V. Kajava 2012). Each structural module of this architecture is a "unit", the assembly of at least three of these building blocks forming a repeat "region" (Di Domenico et al. 2014b). TR protein classification is based on repeat unit length (Andrey V Kajava 2012), which can vary from one or two residues in crystallites (class I) to more than 50 residues in beads-on-a-string (class V), TR proteins built from the repetition of small globular domains (Andrey V Kajava 2012). The middle ground comprises elongated (class III) and closed (class IV) structures, but it is dominated by the presence of a subtype of elongated structures, called solenoids (B. Kobe and Kajava 2000; A. V. Kajava 2001). Mainly due to stabilizing intra-unit short-range interactions, these proteins can be extended and refolded when subjected to a mechanical stretch force (Kim et al., 2010). In addition, they easily tolerate insertion of new units and possess an easily tunable horseshoe shape. These exceptional properties render them very efficient for protein-protein interaction (Andrade et al. 2001), and accounts for their widespread in cellular pathways. There has been an increasing interest in TR proteins, and solenoids in particular, over the last few years, mainly due to their relevance in health (Fournier et al. 2013; de Wit et al. 2011) and for engineering applications (Grove, Cortajarena, and Regan 2008; Höcker 2014; Brunette et al. 2015). However, this class of proteins still belongs to the "dark matter" of the protein universe being characterized by non-canonical sequence-structure relationships. Indeed, solenoid sequences evolve quickly while maintaining their fold, hampering detection by traditional methods for sequence analysis. The same holds for modeling and functional characterization, which usually relies on well conserved sequence features. As a result, specialized methods were built for the identification of non-globular proteins (Pellegrini 2015). Some strategies are sequence-based, and usually rely on self-comparison, pattern recognition or complexity measurement (Pellegrini 2015). Other approaches try to recognize TR proteins based on the modularity of

their 3D structure (Abraham, Rocha, and Pothier 2008; Sabarinathan, Basu, and Sekar 2010; Walsh et al. 2012). RepeatsDB (Di Domenico et al. 2014b) represents the state-of-the-art for the annotation of tandem repeat. The database adds to the typical classification a subclass level based on secondary and tertiary structure features. Existing methods for TR protein identification do not deal with the TR structures classification problem, which was based on manual assignment in RepeatsDB. The database includes manually curated annotation of a subset of entries. This includes the position of each repeat unit and consequent identification of sequence repeats. This particular problem was addressed by few automatic methods. ConSole (Hrabe and Godzik 2014) exploits the modularity of protein contact maps and TAPO (Do Viet, Roche, and Kajava 2015) the periodicities of atomic coordinates and other types of structural representation. Both are available through a web server interface that allows the user to evaluate one protein at a time. The automatic identification of units inside a TR protein structure allows to scale up this type of information.

The newly available data could be a powerful tool to understand TR evolution and to assess conservation at the sequence level. Furthermore, the collection of an “alphabet” of TR units can be useful for protein engineering applications (Brunette et al. 2015). Here we present a new Repeat Protein Unit Predictor (ReUPred) for the classification and identification of repetitive elements in TR proteins. ReUPred outperforms the state of the art methods (TAPO and ConSole) for the unit identification problem and provides subclass assignment for the 87% of the test set with 89% accuracy.

A new metric has been adopted to evaluate correct unit identification considering both phase and unit length. Parameter optimization has been performed manually exploiting a curated dataset from RepeatsDB.

Table 1. Current RepeatsDB annotation of solenoid proteins

Class	Units	Detailed	Classified	Predicted
β	367	41	128	
α/β	180	19	70	
α	388	48	875	
Total	935	108	1,073	7,948

Units lists the number of single defined repeat units. Detailed proteins have the unit position identified manually. Classified are those protein for which the subclass assignment is known, including “manually” and “by similarity”. Predicted proteins are not yet classified.

This work concentrates on the solenoid proteins since it is one of the most abundant class of tandem repeat proteins in nature (A. V. Kajava 2001).

RepeatsDB (Di Domenico et al. 2014b) represents the state-of-the-art for the annotation of tandem repeat proteins. RepeatsDB provides the start and end position of repetitive units only for a small subset (“detailed”) which are manually annotated. Another group of proteins is provided only with the manual classification or by sequence similarity. The rest of the structures (“predicted”) lacks any classification and each represents the majority of the data. The aim of ReUPred is to extend the detailed annotation for all classified proteins and possibly for all predicted repeats. Table 1 summarizes the RepeatsDB classification for solenoids.

6.2.3. Methods

ReUPred is a predictor for the classification of a tandem repeat proteins and identification of the composing repeat units. Its input is a target protein structure and a structural repeat unit library (SRUL). The output is a list of fragments corresponding to the predicted unit positions in the structure and the class assignment according to the RepeatsDB

definition (Di Domenico et al. 2014b). In this work, only solenoid repeat proteins (i.e. classes III.1 to III.3 in RepeatsDB) have been considered as they represent the most abundant class of repeat proteins in nature (A. V. Kajava 2001). The algorithm explores iteratively the input structure using a template library using a *divide and conquer* strategy to improve both accuracy and speed, requiring on average ca. two minutes on a standard laptop. ReUPred was optimized by filtering SRUL, fine tuning parameters to choose the best alignment and detect insertions between units as well as identifying separated repeat regions in the input protein. Each step is described in the following.

6.2.4. SRUL

The Structural Repeat Unit Library (SRUL) constitutes a fundamental part of the ReUPred input and represents the conformational space and diversity of bona fide repeat units. It has been generated by extracting all structural unit fragments from the “detailed” solenoid proteins in RepeatsDB. after filtering units shorter than 10 residues and larger than 90. After filtering units shorter than 10 residues and larger than 90, the solenoid SRUL is composed of 916 structural unit fragments from 108 different proteins and non-redundant at the sequence level. After clustering the sequences with CD-HIT (Fu et al. 2012) at 40% identity cutoff, 531 clusters are obtained. The largest cluster contains 17 units from 5 proteins and the others have less than 10 units each. From the structural point of view, SRUL is biased towards α-helical units. All-against-all structure similarity has been measured by TM-Align (Zhang and Skolnick 2005). Clustering at 0.6 TM-score generates 362 clusters, where the majority of α units (319) fall inside a single cluster.

6.2.5. ReUPred algorithm

The algorithm exploits evolutionary history of tandem repeat proteins. Solenoid units have been demonstrated to evolve from a single representative units to multiple copies through repeated duplications (A.

K. Björklund, Ekman, and Elofsson 2006)(Å. K. Björklund, Ekman, and Elofsson 2006). Units of a solenoid protein show a different degree of similarity which is strongly correlated to the distance from the middle of the repeat region. This is consistent with the observation that units at the edges are more degenerated (E M Marcotte et al. 1999). ReUPred exploits this knowledge and tries to mimic evolution. The objective is to predict adjacent units, i.e. to minimize the number of residues between predicted flanking units and to obtain at least three repeated elements. This is important since in known RepeatsDB solenoid structures, insertions of non-repeat fragments are rare and mostly observed inside and not between units.

See Figure 1 for a schematic description.ReUPred uses an iterative *divide and conquer* approach. Each iteration corresponds to a structural search, i.e. structural alignment of the query structure against all SRUL elements to identify a unit. The predicted unit corresponds to the aligned region in the query. At each cycle the algorithm forks (divide). Two new input structures are created, corresponding to the N- and C-terminal flanking fragments of the predicted unit and two new cycles (structural searches) are performed. After the first cycle, i.e. after the “master” unit is found, SRUL is no longer used. Instead, a new ad hoc library is created on the fly. At the beginning of the second cycle only the “master” unit populates the ad hoc liberary and all newly predicted units are included for search in the following cycles. The algorithm stops when the entire input protein is consumed, i.e. new input fragments are too short, or the structural search does not provide any new valid alignment. At this point the predicted units are collected and evaluated together (conquer). If the result does not satisfy a set of rules, the structural alignment filters for the “master” unit are relaxed and the entire iterative part is repeated from the beginning for up to four increasingly relaxed iterations. This strategy allows to predict both easy and difficult cases automatically. A valid solution for ReUPred is obtained when at least three units are found and their proximity in sequence is ensured by at least one of two simple rules to measure unit proximity: (i) the total number of gaps between units is less than 40 residues, (ii) the number

of non-adjacent units divided by the total number of predicted units is less or equal to 0.25.

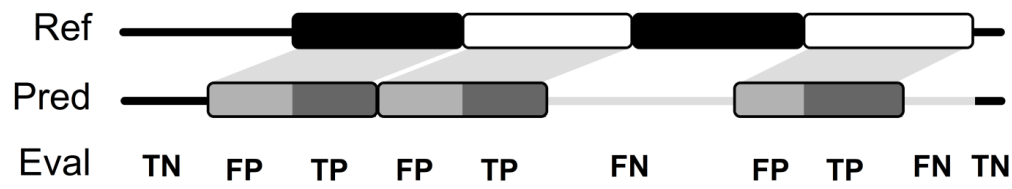


Fig. 1. Schematic overview of the ReUPred algorithm. The input is a PDB structure (PDB 1IQ1, chain C) and the output is a list of unit positions and the predicted subclass. In the structures the “master” unit is in red and the “secondary” units in green and yellow.

Replacing the original SRUL with an ad hoc library from the second cycle onwards improves both computational cost and accuracy. SRUL is very big, with 997 unit templates. Instead, the ad hoc liberary reaches the maximum size at the end of the algorithm and corresponds to the number of predicted units, drastically reducing the number of structural alignments. On the other hand, using only units from the query structure itself increases accuracy as units of the same protein are structurally more similar to each other than units from other proteins (data not shown). The class assignment is provided by simply reporting the classification assigned to the first “master” unit identified from SRUL.ReUPred accuracy strongly depends on the quality of the structural alignments at each cycle. In particular, it is very important to correctly predict the first “master” unit because errors propagate. Alignments have to abide a set of rules and constraints that are much more stringent for the “master” search compared to successive cycles. Structural alignments are calculated using TM-Align (Zhang and Skolnick 2005), filtering by TM-Score, RMSD, alignment length and number of gaps. Tables 2 and 3 list all cutoff values for the cascaded four runs used to select valid alignments for the “master” and “secondary” units, executed on cascade until a valid solution is found.

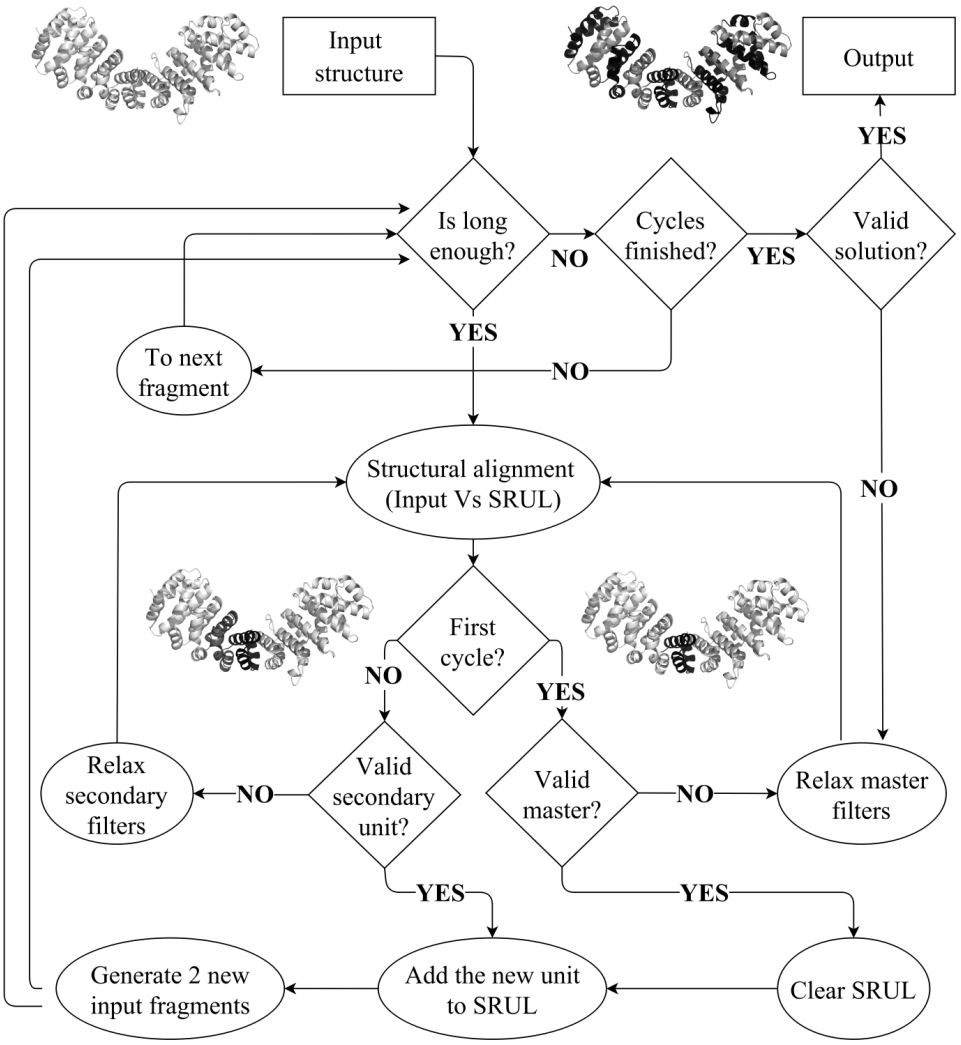


Fig. 2. Unit prediction evaluation. In panel A two wrong predictions (gray) and the reference (blue). In panel B the definition adopted in this work to define correct (red) and wrong (gray) predicted residues.

Table 2. Structural alignment constraints for the “master” unit.

Iteration	TM-Score	RMSD (A)	Alignment (residues)	Unit gaps (%)
1	≥ 0.52	≤ 1.6	> 21	< 10
2	≥ 0.47	≤ 1.9	> 17	< 20
3	≥ 0.30	≤ 2.5	> 16	< 50
4	≥ 0.23	≤ 3.0	> 14	< 50

TM-Score and RMSD are the same provided by TM-Align. Coverage and gap are calculated as described in the manuscript. Different columns correspond to different algorithm runs that are performed on cascade until a valid solution is found.

Table 3. Structural alignment constraints for the “secondary” units.

Iteration	TM-Score	RMSD (A)	Alignment (residues)	Unit gaps (%)	Length ratio (%)
1	≥ 0.35	≤ 1.8	≤ 1.20	< 40	≥ 70
2	≥ 0.30	≤ 2.0	≤ 1.15	< 40	≥ 70
3	≥ 0.30	≤ 2.5	≤ 1.15	< 40	≥ 70
4	≥ 0.30	≤ 3.0	≤ 1.10	< 50	≥ 70

Columns are as in Table 2. The length ratio is calculated as the unit length divided by the length of the first “master” unit.

6.2.6. Performance evaluation

Evaluating the quality of a prediction is not easy as it is necessary to define a metric to measure the correct matching of the predicted units (reference). Figure 2 shows a comparison between two predictions (gray) with the RepeatsDB reference (blue). Both predictions are wrong as the first has a wrong phase (all units are shifted forward) while the second predicts units with the correct phase but double size. A new strategy was implemented that takes into consideration both aspects when evaluating a prediction by measuring the overlap of predicted units with each reference unit. In the second panel of Figure 2 another example is reported and correct and wrong predicted residues are highlighted in red and gray respectively. To perform this type of evaluation correctly, before generating the confusion matrix, it is necessary to match predicted units with the reference counterpart using a maximum overlap criterion. For example in panel B of the figure the

last predicted unit is compared with the last unit of the reference since the overlap is greater compared with the preceding unit.

In the example of Figure 2 (panel B) the predicted true positive residues correspond to the red area of the units whereas the true negatives correspond to red fragments outside the repeat region. False positives are gray areas inside the units, while false negatives are represented by the gray segment, i.e. structure fragments with units in the reference but not predicted.

Filtering the parameters for filtering structural alignments has been performed manually, maximizing test set coverage. The number of repeat proteins for which a valid output is provided, and prediction accuracy, i.e. correct unit position assignment. The training set was built considering all solenoid proteins in RepeatsDB for which unit annotation is provided ("detailed" entries). Since SRUL has been generated from the same protein set to evaluate ReUPred ability to generalize, all units coming from the target itself and all similar units were removed at each step from the SRUL. Template similarity was measured at sequence level by setting a cutoff of 30% of sequence identity.

For the unit centric evaluation, true positives (TP), false positives (FP), true negatives (TN) and false negatives (FN) are calculated as described in Section 2.5. For classification, given a solenoid subclass TP is the number of proteins with correct assignment, FN are proteins assigned the wrong class, FP are class assignments to wrong targets. TN is always zero since the test set contains only classified proteins. For all evaluations, the measures recall (or sensitivity; TP/(TP+FN)), precision (TP/(TP+FP)) and accuracy ((TP+TN)/(TP+FP+TN+FN)) are used. ReUPred is compared to the methods TAPO (Do Viet, Roche, and Kajava 2015) and ConSole (Hrabe and Godzik 2014). TAPO predictions have been generated from the web server (default parameters) considering only the first solution. ConSole predictions were generated locally by the stand-alone software (default parameters). The RAPHAEL period is provided in the RepeatsDB entry metadata. For all evaluations, ReUPred has been benchmarked after removing from SRUL units

coming from the test protein or structurally similar units (see Section 2.5). The comparison with TAPO and ConSole has been performed on the set of proteins for which all methods predict at least one unit, i.e. 89 out of 108 proteins.

Fig. 3. ReUPred unit prediction for Plakophilin-1 (PDB code 1XM9, chain A). The structure is shown in cartoon representation in the top part with the schematized sequence below. Predicted units are represented in black and grey. Dashed lines represent missing residues in the PDB file (residues 388-396 and 481-508). The N- and C-terminal residues flanking the missing residues are shown as spheres in the structure.

Three different datasets have been used. The first has been generated from the “detailed” RepeatsDB entries (108 proteins) and represents the reference for unit prediction evaluation. Another set with all “classified” and “by similarity” entries (1,075 proteins) has been used to test the ability to automatically classify repeat proteins and compare unit length prediction with RAPHAEL (Walsh et al. 2012). A third dataset has been used to test the discrimination of negative examples, i.e. non-repeat proteins. In this case, the dataset is from the RAPHAEL paper (Walsh et al. 2012), i.e. 247 non-solenoid proteins with different topologies and no detectable sequence similarity.

ReUPred was developed to predict both unit position and classify repeat proteins in order to automate the time-consuming manual annotation

process of “detailed” annotation in RepeatsDB. See Figure 3 for an example on Plakophilin-1. Before benchmarking the main novel features, it is worthwhile to investigate whether ReUPred is able to correctly discriminate real repeats from non-repeat proteins. For this purpose, it has been compared with RAPHAEL (Walsh et al. 2012) on the original dataset with 247 non-repeats. ReUPred correctly provides no prediction in 238 cases, corresponding to a specificity of 96.36%. This is only marginally lower than RAPHAEL at 97.2% on the same dataset. A better result could be obtained for ReUPred by setting a stronger filter on the last step of the algorithm, but that would affect coverage on the positive dataset. Even though ReUPred was designed to predict unit positions in tandem repeat proteins, this result demonstrates that the tool is also effective in discriminating repeat/non-repeat proteins.

Table 4. ReUPred ability to predict solenoid classification

Class	Recall	Precision	F-Measure	Accuracy
All-β	0.81	0.74	0.78	0.63
Mixed α/β	0.55	0.65	0.60	0.43
All-α	1.00	0.99	1.00	0.99
Total	0.94	0.94	0.94	0.89

6.2.7. Repeat classification

ReUPred predicts units and fine classification for 83% (893 proteins) of the test set. The class assignment is obtained by simply transferring this information from the master unit found in SRUL. This approach has been proven to be effective as shown in Table 4. ReUPred works very well for the α class (III.3 in RepeatsDB). Instead, it is more difficult to correctly assign α/β and β examples. The low recall indicates that the

cause of the problem is detecting units that do not have a good template in SRUL.

Table 5. Unit prediction evaluation

Class	Method	Recall	Precision	F-Measure	Accuracy
All-β	TAPO	0.47	0.59	0.53	0.47
	ConSole	0.39	0.69	0.50	0.46
	ReUPred	**0.62**	**0.64**	**0.64**	**0.56**
Mixed α/β	TAPO	0.66	0.70	0.68	0.59
	ConSole	0.62	0.69	0.66	0.57
	ReUPred	**0.84**	**0.84**	**0.84**	**0.78**
All-α	TAPO	0.64	0.78	0.70	0.57
	ConSole	0.50	0.74	0.59	0.46
	ReUPred	**0.74**	**0.79**	**0.74**	**0.62**
Total	TAPO	0.58	0.70	0.64	0.53
	ConSole	0.48	0.71	0.58	0.49
	ReUPred	**0.71**	**0.75**	**0.73**	**0.62**

Performance evaluation is reported for all RepeatsDB solenoid structures (All) and for the three subclasses separately (β, α/β and α)

This is an important result, as it indicates which RepeatsDB entries are worth manually annotating at the “detailed” level to improve ReUPred sensitivity and SRUL representation of the repetitive structural element universe. Low precision for β and α/β classes is generated by a high number of false positive assignments. Looking at the data in detail, we found some ambiguous class assignments. E.g. PDB code 3ZYI, chain A, is annotated as α/β solenoid in RepeatsDB but there are no helix elements except for a small fragment (residues 309-318) which is not repeated in the units. Since ReUPred predicts the class by transferring annotation from SRUL, if a SRUL element is misclassified the error propagates. ReUPred could be very useful to guide the manual refinement of RepeatsDB class annotations.

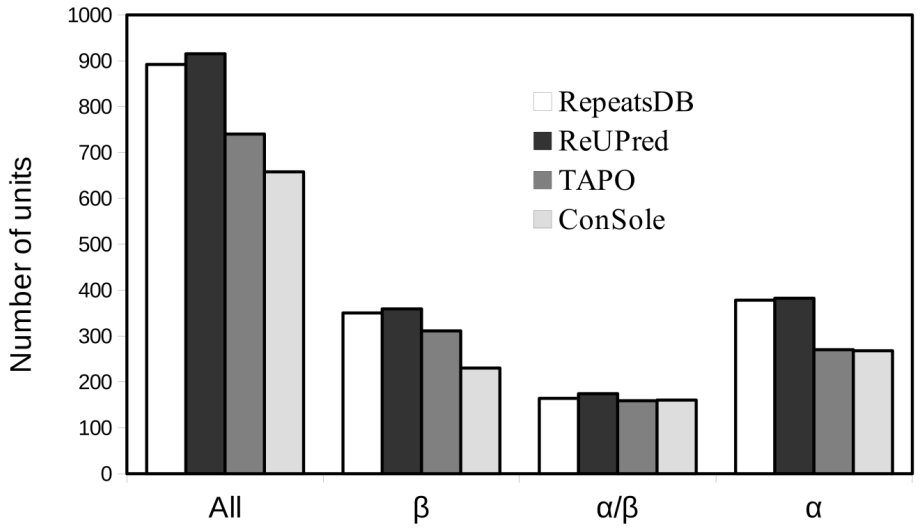


Fig. 4. Number of predicted units on the RepeatsDB detailed dataset. The manually curated reference (RepeatsDB) is shown next to the three prediction methods. ReUPred predicts more repeat units than the other two methods.

6.2.8. Unit prediction accuracy

ReUPred has been evaluated for unit prediction using the metric described in methods section, i.e. penalizing predictions with a wrong phase or/and a wrong length. Table 5 shows a comparison with TAPO and ConSole in terms of predicted repeat residues on the "detailed" solenoid entries in RepeatsDB. Results are reported for each of the three main solenoid classes and for all proteins together. ReUPred always outperforms the other methods for all evaluation measures. In particular, the greatest improvement is observed for the α/β subclass, with an increase of 19% accuracy compared with TAPO. The high accuracy for this class can be explained by the fact that mixed α/β units represent more structurally complex elements compared to all-α units. More information is coded in the structure unit, making it easier to discriminate wrong structural alignments. On the other hand, the most problematic subclass is all-β. Both recall and precision are lower for all methods compared with other subclasses. This may be explained by the fact that β solenoid units are more degenerated in the same protein than other solenoids and present a greater structural diversity with many

insertions (data not shown). Moreover, they are shorter compared with all-α, generating worse structural alignments.

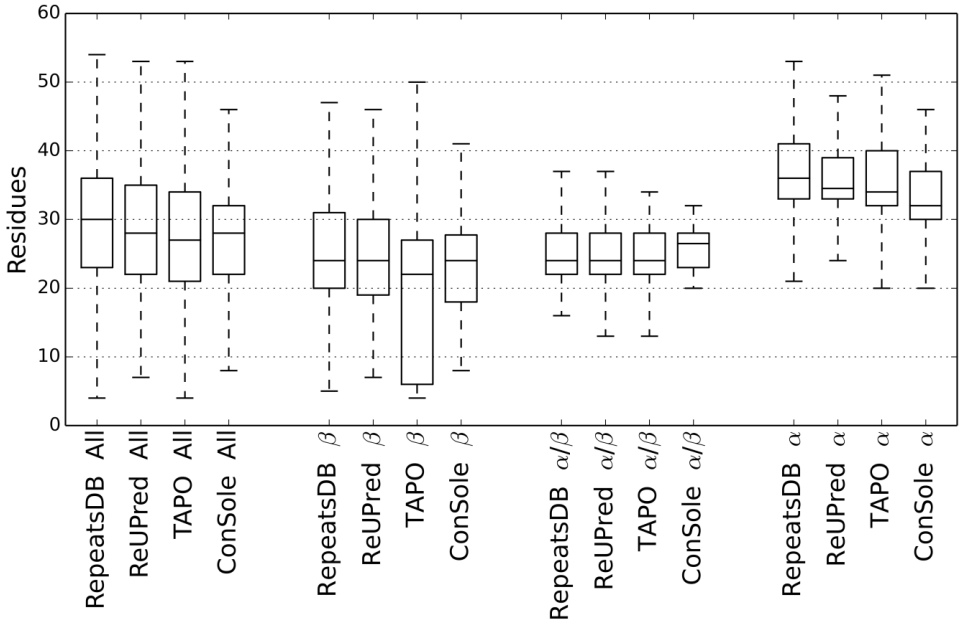


Fig. 5. Repeat unit periodicity box plot distribution on the RepeatsDB detailed dataset. The manually curated reference (RepeatsDB) is shown next to the three prediction methods.

In addition to evaluating repeats annotations at the residue level, it is of interest to benchmark repeat units and their length distributions. Figure 4 shows the number of repeat units being identified by each method. Here again, ReUPred predicts more units than the other two methods. Both ConSole and TAPO generate units with the same size for a given structure and this may limit their ability to deal with insertions in solenoid proteins. ReUPred may therefore be better able to adapt to the irregular aspects of solenoid repeats.

Figure 5 shows a box plot for the distribution of predicted repeat periodicities against the RepeatsDB reference set. The median repeat length and standard deviations of ReUPred are very similar to the reference definition and on average match better than TAPO and ConSole. TAPO appears to under-predict the repeat length in β

structures, probably because it also uses sequence information. ConSole on the other hand appears to have more difficulty with α-helices.

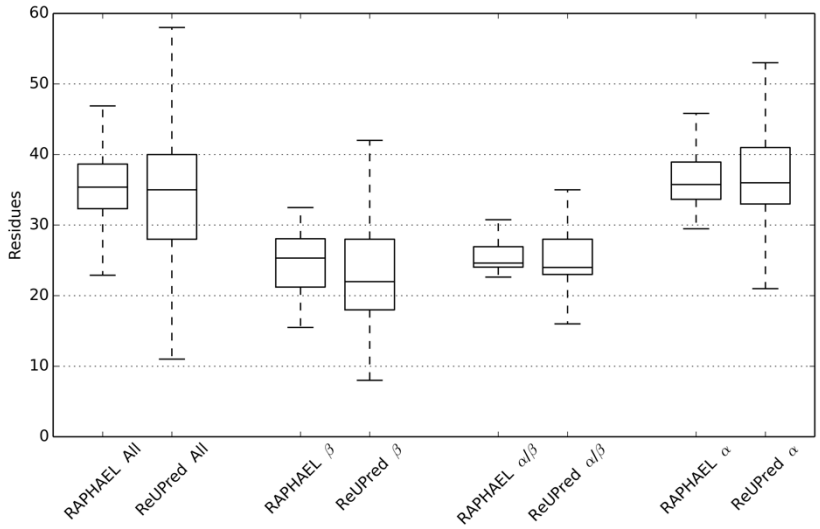


Fig. 6. Large-scale periodicity predictions on the RepeatsDB classified dataset. The original RAPHAEL periodicities are compared to ReUPred unit lengths as box plot.

6.2.9. Expanding the universe of known solenoids

Given the good performance of ReUPred for its intended purpose, i.e. classifying solenoid repeats and annotating their component units, it can be used to automatically expand the knowledge contained in RepeatsDB. The first step consists in establishing the baseline against the existing RAPHAEL annotations on the "classified" dataset. This contains annotations for solenoid class and predicted average repeat length. Since this dataset does not provide unit annotation, the simplest way to evaluate the performance is to compare the length of the predicted units with the repeat period predicted by RAPHAEL. This is the number of residues for which the symmetry signal is maximized, generating a single period for each protein. This is a big limitation, as it does not reflect the real situation where unit sizes vary inside a protein due to insertions which are frequent in solenoids. In particular, it is very

relevant for the all-β class where almost all proteins have insertions. Figure 6 compares the distribution of ReUPred predicted unit length and RAPHAEL period for each solenoid class. Overall, both are very similar, with ReUPred having a wider range of periodicities as it is able to recognize irregularities in single repeat units. Only the distributions for all-β repeats differs more markedly. This class contains many structures with insertions which RAPHAEL struggles to summarize in a single fixed periodicity.

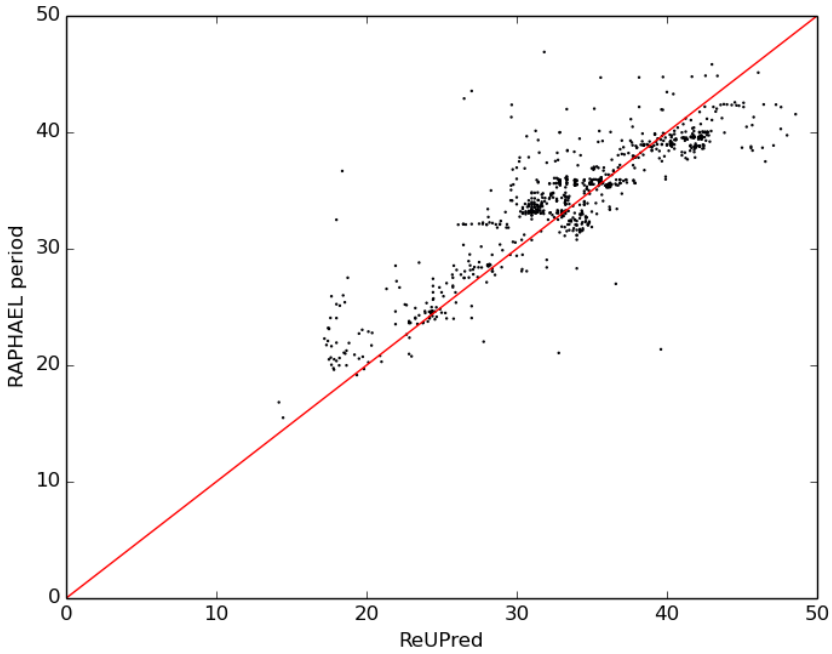


Fig. 7. Scatter plot of RAPHAEL and ReUPred periodicities on the RepeatsDB classified dataset. RAPHAEL produces a single periodicity per protein, whereas all predicted units were considered for ReUPred.

The scatter plot in Figure 7 shows the correlation between the RAPHAEL period and ReUPred mean unit length calculated on each predicted protein. The two methods correlate strongly, with a Pearson correlation coefficient of 0.88 (P-value = $4.59*10^{-290}$). On average, ReUPred predicts shorter units than the RAPHAEL period, 33.7 (SD 6.5) and 34.2 (SD 5.3) residues respectively. When the RAPHAEL period is much larger (extreme points above the diagonal), ReUPred wrongly predicts two units instead of a single unit which would better represent the repetitive symmetry (e.g. PDB code 3L3F, chain X). For opposite

cases happens the contrary, i.e. ReUPred predicts a pair of units as a single element (e.g. PDB code 3PET, chain A).

In order to expand the annotation in RepeatsDB, ReUPred has been used to predict all repeat units for “classified” RepeatsDB solenoids. Since no comparison no structural validation is possible, we chose to compare the annotation to Pfam. Figure 8 shows the very substantial increase in annotations both in terms of for bona fide solenoids proteins and especially in the number of identified repeat units. The latter yields an increase of an order of magnitude compared to state-of-the-art sequence-based annotation in Pfam.

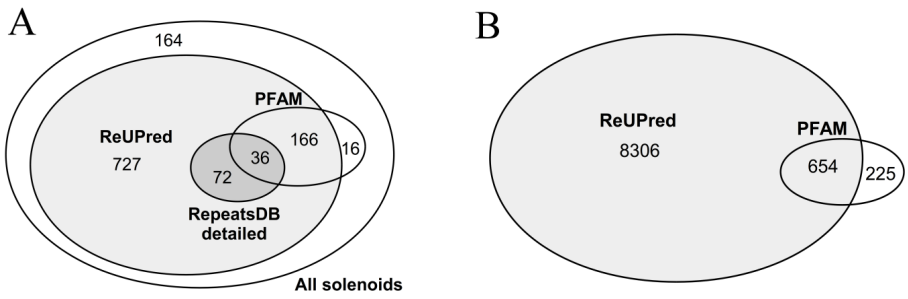


Fig. 8. Venn diagram of available annotations for RepeatsDB classified dataset. (A) compares proteins with bona fide solenoid assignments. (B) shows the number of annotated repeat units in the dataset. The total number of repeat units in the dataset is unknown. ReUPred is able to increase the annotation by an order of magnitude in both cases.

6.2.10. Conclusion

Tandem repeat unit prediction and classification are difficult problems currently tackled by expert manual curation which at the time of writing is available in RepeatsDB for only 3% of the total putative repeat protein structures. ReUPred provides both the prediction of the repetitive units and a fine classification up to the subclass level is the RepeatsDB classification scheme. The algorithm works by exploiting a structure repeat unit library (SRUL) and an iterative exploration of the input structure. While we tested the performance on the solenoid class, the method also works for other repeat types. ReUPred has been compared with other state-of-the-art methods, TAPO and ConSole, adopting an evaluation metric which takes into consideration both phase and size of

the predicted units. Testing on a manually curated data set obtained from the “detailed” RepeatsDB entries, ReUPred achieved the highest accuracy for all type of solenoids (β, α/β and α) with an overall increase of 9% over TAPO and 13% over ConSole. To provide an extended evaluation, a larger dataset has been created by collecting RepeatsDB entries for which only the classification is available without unit annotation. With this in hand, it was possible to test ReUPred ability of classifying solenoid structures and the correlation with periods predicted by RAPHAEL. ReUPred extended unit annotation and classifies at the subclass level. Almost all solenoids have high precision and accuracy. Moreover, the average unit length predicted by ReUPred strongly correlates with RAPHAEL, confirming the high quality of the predictions. Mixed α/β unit diversity is underrepresented in SRUL compared to the α and β classes. This means that improving SRUL could correspond to a better recall and therefore an even higher accuracy. ReUPred has also the ability to detect the unit diversity inside a given target protein. It recognizes fragment insertions that are not part of the repeat elements. This work demonstrates that repeat protein annotation can be faced by repetitive template based structural searches. Moreover, it shows that this approach can be applied reliably on a large scale, i.e. over all uncharacterized RepeatsDB entries, unveiling new scenarios for the analysis of the entire repeat protein universe.

It is also able to discriminate with good accuracy between repeat and non-repeat proteins, making ReUPred a good candidate to replace RAPHAEL in the detection of proteins containing tandem repeat domains

6.3. The Victor C++ Library for Protein Representation and Advanced Manipulation

Authors: Layla Hirsh, Damiano Piovesan, Manuel Giollo, Carlo Ferrari, and Silvio C.E. Tosatto

Journal: Bioinformatics (2015) 31 (7): 1138-1140.

doi: 10.1093/bioinformatics/btu773

6.3.1. Abstract

Motivation: Protein sequence and structure representation and manipulation require dedicated software libraries to support methods of increasing complexity. Here, we describe the VIrtual Constrution TOol for pRoteins (Victor) Cþþ library, an open source platform dedicated to enabling inexperienced users to develop advanced tools and gathering contributions from the community. The provided application examples cover statistical energy potentials, profile–profile sequence alignments and ab initio loop modeling. Victor was used over the last 15 years in several publica- tions and optimized for efficiency. It is provided as a GitHub repository with source files and unit tests, plus extensive online documentation, including a Wiki with help files and tutorials, examples and Doxygen documentation.

Availability and implementation: The Cþþ library and online documentation, distributed under a GPL license are available from URL: http://protein.bio.unipd.it/victor/.

6.3.2. Introduction

Structural bioinformatics methods require valid software libraries to represent and manipulate proteins efficiently. A number of widely used tools have been developed over the years to visualize proteins, e.g. Chimera (Huang et al., 2014), Swiss-PdbViewer (Guex et al., 2009), MolIDE (Canutescu and Dunbrack, 2005) and VMD (Humphrey et al., 1996) to name a few. Software libraries to ma- nipulate proteins efficiently provide basic data representation and more advanced functionality with a different focus each. ESBTL (Loriot et al., 2010) is mainly a Protein Data Bank (PDB) file parser. Biskit (Gru¨nberg et al., 2007) additionally provides functionality for analysis of molecular dynamics simulations, while PTools (Saladin et al., 2009) focuses on molecular docking. OpenStructure (Biasini et al., 2010) places more

attention on structure visualization and en- ergy calculation. The latter is also supported by MSL (Kulp et al., 2012) and Tinker (Shi et al., 2013), while BALL (Hildebrandt et al., 2010) in addition provides many advanced optimization algorithms.

Finally, StrBioLib (Chandonia, 2007) extracts sequence information from the protein structure and can be used as an interface to several available third-party tools.

The critical assessment of techniques for protein structure pre- diction (CASP) series of experiments (Moult et al., 2014) demon- strates that structure prediction is increasingly becoming an engineering problem, where sophisticated methods have to be com- bined into extensive pipelines to provide state-of-the-art results (Khoury et al., 2014). This has raised the barrier for entry into the field to a point where little new developments are possible, consider- ing that most software libraries used in CASP are proprietary and not available as open source. Here, we propose the open-source VIrtual Constrution TOol for pRoteins (Victor) Cþþ library as a way to mitigate this problem. Victor is both an efficiently designed Cþþ library, able to manipulate protein structures with minimal computing time, and a collection of advanced components for pro- tein sequence and structure manipulation. In particular, Victor provides three sample applications: profile–profile sequence align- ments (Wang and Dunbrack, 2004), statistical potentials (Tosatto, 2005) and loop modelling (Tosatto et al., 2002). Each of these three applications has been extensively described in the literature and is beyond the scope of this article. To the best of our knowledge, nei- ther is available as an open-source Cþþ library yet. Profile–profile sequence alignments, in particular, have been widely used to im- prove target-template alignment in CASP (Kryshtafovych et al., 2014). Victor is composed of >60 000 lines of code and still expand- ing as it is used in the main author's teaching. It was developed in- house over the last 15 years with the contribution of tens of devel- opers and has reached a high level of maturity. Victor is released to provide a platform for

contributions from the interested community. It provides extensive online material in the form of a Wiki with help files, tutorials, Doxygen documentation and a list of applications built using Victor can be accessed from the URL: http://protein.bio. unipd.it/victor/. The actual GitHub repository with Cþþ source files, a precompiled Ubuntu 64-bit version and unit tests are avail- able from URL: https://github.com/BioComputingUP/Victor.

6.3.3. Core library

The Victor Cþþ library currently contains two components for data representation and manipulation in separate directories: tools and Biopool. Tools provide basic manipulation methods, e.g. vector co- ordinates and file I/O. The core of the library is provided by the Biopool module, which defines all relevant data structures and algo- rithms to represent protein structures and manipulate them at a higher level of abstraction. The core data structures were carefully developed using design patterns (Gamma et al., 1995), to provide an elegant and simple, yet powerful set of Cþþ classes. To allow the simple manipu- lation of protein structure through the more intuitive torsion angles, automating low-level geometric transformations, atom positions are coded both explicitly in 3D coordinates and as a position relative to the previous atom on a graph structure. This ensures consistency in the structure, while allowing the programmer to change the protein conformation rotating a torsion angle with a single line of code. Computational efficiency is guaranteed by updating the corresponding Cartesian coordinates only when necessary. All low-level geometrical transformations remain transparent to the user. Biopool is able to read properly all existing PDB files. Additional tools are also pro- vided, such as protein secondary structure automatic assignment with an ad hoc implementation of the original DSSP algorithm (Kabsch and Sander, 1983). Extensive online documentation allows the inter- ested programmer to learn how to manipulate the Biopool data structures.

6.3.4. Applications

The Victor library provides three main examples to demonstrate the range of possible applications, which are included as separate sub-directories: Energy, Align and Lobo. Extensive documentation, including detailed tutorials, is provided online to allow users to be- come familiar with the software and build on existing knowledge. Energy contains everything that is necessary to develop statistical potentials to evaluate protein structures. Two sample implementa- tions of published methods included in the library, FRST (Tosatto, 2005) and TAP (Tosatto and Battistutta, 2007), can serve as a guide to develop additional methods. Both are contained in the Energy subdirectory and functioning code is provided both to generate the statistical potential itself as well as to use it on a PDB structure to calculate the potential energy. The interested user can thus easily de- velop additional statistical potentials. The Align directory provides basic sequence alignment algorithms (Tosatto et al., 2006) augmented with secondary structure element (Fontana et al., 2005). Many different profile–profile scoring schemes (Wang and Dunbrack, 2004) are implemented, which have been ex- tensively used in CASP to detect remotely homologous protein se- quences. Code is also provided for variable gap penalties with additional terms for sequence to structure fit (Madhusudhan et al., 2006) and advanced weighting schemes such as PSIC (Sunyaev et al., 1999). Alignment parameters have been extensively benchmarked and the default parameters are optimized for performance.

Last but not least, the Lobo directory contains an application of ab initio loop modeling using a fast divide and conquer algorithm (Tosatto et al., 2002). This makes extensive use of the functions to construct novel amino acids and manipulate the protein structure lo- cally, providing sample code for more complex structural manipula- tions. It can easily be extended for ab initio structure prediction in combination with statistical potentials as target function.

6.3.5. Conclusions

The Victor library is an open source project devoted to the structural bioinformatics community. It provides a unique combination of methods for sequence and structure manipulation. Expansion is on- going both through in-house development, as it is the basis for sev- eral more recent publications [e.g. RING (Martin et al., 2011) and NeEMO (Giollo et al., 2014)], and as part of the author's teaching activities, which include software development projects for students. We hope that the Victor library will contribute towards an easier de- velopment of advanced methods for structural bioinformatics.