Proteins are often called the building blocks of life, however they are much more than that. Whole function of life as we know it hinges on proper protein behaviour. Still, there remain many unknowns about the specifics of how the genomic sequence enforces folding to a particular structure, why is this shape necessary to properly fulfil a given function. In particular, one of the most mysterious elements in the process outlined above is structural kink that has been found in approximately 2% of known proteins structures (as indicated by KnotProt (Jamroz et al., 2014) and LinkProt (Dabrowski-Tumanski et al., 2016) databases) – a non-trivial topology of the backbone of the protein chain. This non-triviality is meant in a quasi-mathematical sense – that is, it relates to knots and links described by knot theory, with some provisions made due to the open nature of protein chains, which we explore further in the thesis.

This dissertation describes various algorithms and methods useful for study of proteins at different levels of organisation (Fig. 1) – from an algorithm for detecting a non-trivial fold in a structure (presented in Chapter 2), through algorithms for multiple alignment of sequences and methods for relating sequence variance with its corresponding structure (Chapter 3), up to tools and databases for knots and links in multiple chains (presented in Chapter 4).

Detection of knot-like folds in biological molecules

Finding knot-like structures in biological molecules is a non-trivial task – most of the polymers found in biology are open chains, and not closed curves as expected by the mathematical knot definition. Thus, in general, when a phrase ’protein knot’ appears, the implicit idea is of a ’common sense’ knot, as one would find on a string – a structure in which pulling on both ends would not result in a straight line. One notable exception are DNA molecules which can be circular, such as plasmids, and can actually be rather easily knotted and unknotted by topoisomerases (enzymes which sole function is allowing the DNA chain to pass through itself). As such, the circular DNA knots and links have been known and studied for the last four decades (Macgregor and Vlad, 1972; Sumners, 1995).

Much more complex topic are knots on open biological polymers – such as proteins, RNA, and chromatin (open DNA chains). Finding them necessitates a more lax approach to the mathematical definition, as the curve of the backbones of those molecules must be closed first, to allow the use of tools of knot theory. Typical approach consists of virtually
Rysunek 1: Schematic representation of the overlap between various topics included in this thesis.

elongating the termini multiple times in random directions, connecting them on a surface of a large sphere around the structure, and calculating the knot type of each resulting closed chain separately. This gives a probability score to each knot type found (an unknot score for Mansfield (Mansfield, 1994)), with the 40% probability cut-off often used to determine the actual presence of a knot (Jamroz et al., 2014).

The most interesting – due to the most diverse nature of chain elements – appear to be proteins. While it can be easily shown, that collapsing a long chain will usually lead to an entangled structure (Levitt, 1976; Némethy and Scheraga, 1977; Skolnick and Kolinski, 1991; Chan and Dill, 1993) and this will in fact represent the most advantageous packing of the polymer, protein folding have been long thought to elude this tendency (Bryant et al., 1974). While folding is generally thought to be guided by hydrophobic collapse, various interactions (both attractive and repulsive) between amino acids that make up the protein chain complicate this process – in particular, a lack of reptative motions of the chain is expected (as the protein chain is not a smooth one). Because of this, the knotting should happen close to its native position in the structure – giving the protein entanglements another
possible classifier: their depth. This depth is defined as the minimum number of residues have to be removed from either end of the chain to untangle it. An often complimentary quantity is ‘tightness’ of a knot, which describes how long is the knot core – minimal knotted fragment of a structure.

Due to the fact that the entanglement cannot slide along the protein chain, the rate limiting step in formation of a knot-like structure is assumed to be the piercing of the loop. Consequently, all the protein entanglements found to date correspond to knots which can be made by just a single passing of the chain through a (potentially multiple times) twisted loop (Sułkowska et al., 2012; Taylor, 2007). For example – while a $5_2$ knot-like structure can be found in a family of deubiquitinases, there are no known structures similar to the $5_1$ knot (see Fig. 2).

![Rysunek 2: Twist knots are knots that can be made with just one threading through a twisted loop.](image)

In Chapter 2, we present knot_pull – a new tool for analysing the topology in open chains, such as proteins, RNA and DNA (chromatin). It was designed to bypass some of the particularities of currently used methods.

Mathematically, knots are embeddings of a closed curve (simply put, a circle) in three dimensional Euclidean space. Knots can be classified according to their complexity, expressed as the number of crossings (or double points) in their projection on a surface (a diagram).

**Definition 1** (Link diagram). An orthogonal projection of a knot or link into a plane that contains a finite number of multiple points (double points with transverse crossing), gives a link diagram $\mathcal{D}$ – an undirected labeled planar graph, which satisfies following conditions:

1. loops are connected components with no vertices (thus disconnected from the rest of the graph);

2. both ends of a non-loop edge lead to one vertex each (possibly the same one), and both are labeled as undercrossing or overcrossing in corresponding vertex;

3. there are two overcrossing, and two undercrossing, incident edges leading to each vertex, and their cyclic ordering alternates under- and overcrossings.
A knot diagram is a link diagram with only one connected component. Vertices of a link diagram are called crossings.

Presently, most software (Tubiana et al., 2018; Lua, 2012; Jamroz et al., 2014) for detection of knots in biomolecules uses the same basic approach:

1. smoothing (simplification) of the chain, to get a curve with the same overall topology, but less crossings in a plane projection;

2. closing the curve on the surface of a large (implicitly – infinite) sphere surrounding the structure – this step can be error prone, as the closures could add additional entanglement to the chain. Thus such closure is usually repeated multiple times, to lessen the impact of occasionally erroneous path.

3. Closed chain is projected on a plane, and a knot invariant (e.g. Alexander polynomial (Alexander, 1928), or HOMFLY-PT polynomial (described below, Equation 1) is calculated. For multiple closures this gives a probability that the structure contains a given type of a knot.

Usually knots are recognised through knot invariants. Simply put, a knot invariant is any quantity which can be defined for any knot, and is the same for equivalent (i.e. isomorphic) knots. However, it is important to note that the reverse does not hold – that is dissimilar knots can have the same invariant (e.g. in case of Alexander polynomial a prime, 8-crossing knot as the same invariant as a composite 6-crossing knot). Most commonly used knot invariants are knot polynomials (Alexander, 1928), which are computed on a given knot diagram (although the choice of a diagram is irrelevant, as per the definition of the invariant). Examples include Alexander polynomial (Alexander, 1928), Jones polynomial (Jones, 1985) and HOMFLY-PT polynomial (Freyd et al., 1990; Przytycki and Traczyk, 1988) (the last one used also for links). Polynomial coefficients are calculated through manipulation of crossing directions (e.g. using skein relations), and, as invariants, encode some properties of a knot.

The HOMFLY-PT polynomial is a generalisation of the Alexander and Jones polynomials, and can be transformed into both given appropriate substitutions. The polynomial is defined using skein relations (Fig. 3), which give a linear relation between knot polynomial values for links which differ by one crossing (skein relations are sufficient to calculate Alexander and Jones polynomials just by recursion).

With link diagrams $L_-, L_+, L_0$ as pictured in Fig. 3, the HOMFLY-PT polynomial can be defined as follows:

$$P_U(l, m) = 1$$

$$lP_{L_+}(l, m) + l^{-1}P_{L_-}(l, m) + mP_{L_0}(l, m) = 0,$$
where $U$ is an unknot, and $l$ and $m$ are polynomial coefficients used to distinguish the invariants of different knots. Some main properties of this polynomial include:

- HOMFLY-PT polynomial of a composite knot is the product of polynomials of its components;
- HOMFLY-PT polynomial can be used to distinguish two knots of different chirality, as $P_K(l, m) = P_{\text{Mirror image}(K)}(l^{-1}, m)$.

We propose to describe the knot type of molecule using the Dowker-Thistlethwaite (DT) code (Dowker and Thistlethwaite, 1983). To obtain this notation for a surface projection of a knot, we start at some arbitrary point of the curve and while moving along the string we number each crossing that we pass. For a proper knot when we get back to the starting point each crossing should be labeled by two values – one odd and one even (Fig. 4). To also take into account chirality of the structure, the crossings can be additionally marked to indicate whether we pass under or over the second line when crossing (which is indicated by a minus sign next to the even value if we add it when going over). The notation can then be shortened, by sorting the label pairs according to the odd value – final order (and positive/negative sign) of the even numbers indicate the knot type. However, it should be noted that this is not an invariant – as one knot diagram can have multiple different notations.

DT code of a structure contains significantly more information about the structure it describes as compared with the knot invariants – precisely because the same knot diagram...
(for a closed chain) can have multiple different codes, corresponding to different starting points, and direction, for numbering. In biomolecules, both of those follow naturally, from the N- to the C-terminus. As the number of crossings in a knot diagram of even a simplified protein can be quite large, we propose an algorithm for reducing the DT code using a sequence of modifications based on the Reidemeister moves (Reidemeister, 1927).

To simplify the task of calculating the shortest DT code of a structure, as well as to help visualise a knotted chain, we propose a new smoothing algorithm for open polymers.

**Modeling protein sequence evolution**

Protein sequences provide the most information when studied in comparison with others. The order of residues by itself is not yet understood enough to provide much data beyond some basic hydrophobic/hydrophilic differentiation (although this gives some intuition about how structurally buried we can expect a given region to be (Callaway, 1994)). Any further characteristics, such as predicted secondary structure elements, or domain organisation, can only be identified in comparison to already known – extracted from already known proteins structures – statistics (in case of secondary structure probability) or patterns (for domains).

All the evolutionary diversity of phenotypes, considered both on micro- and macroscopic levels, has its roots in just a handful of processes of molecular evolution. Large scale genomic rearrangements – such as gene duplications – are crucial for new proteins to be able to emerge. A duplicated gene has more freedom to mutate, and as long as one of the copies behaves as expected, the other may temporarily (in terms of evolution) lose or change its functions. However, the actual engine of persistent change\(^1\) are the mutations (including insertions and deletions, known together as indels) that affect single nucleotides. When looking at a single nucleotide in a protein-encoding gene, changing the base can start a cascade of changes throughout the central dogma of molecular biology and beyond. If the mutation is not silent – that is the amino acid encoded by the nucleotide triad that changed will be different – the sequence of the protein will change. This can result in a direct loss of function, if e.g. the residue was important in ligand binding, or a change to the structure which can then impact the function of the protein or even prohibit correct folding.

**Sequence alignment**

The most straightforward way of comparing two sequences, represented as strings, of same length is to calculate their percentage identity, or some form of edit distance (number of positions which differ). Those metrics can be extended to sequences of different length by finding their 'alignment' which maximizes the identity/minimizes the distance. Alignment

---

\(^1\)Here we only account for mutations to the genome of the organisms – so those that can be passed on to the offspring.
here is the association of corresponding positions from each sequence – in practice writing one of the sequences below the others, adding gap symbols signifying indels to one or both, in a way that optimizes the column-wise score. Alignments are made to satisfy at least one of the following criteria (Claverie and Notredame, 2006):

- evolutionary similarity, where aligned residues hail from the same residue in the ancestral protein;
- structural similarity, where aligned residues are placed in approximately the same position in the protein structure;
- functional similarity, in which aligned residues fulfil the same role in the function of the protein.

For related sequences all three criteria are usually highly convergent, but none can be universally calculated outright from the sequence. Additionally, three sequences with the same number of conserved positions and pairwise identity percentages may still differ in similarity. The assumption that some changes between amino acids are more likely to be successful – or rather mutation between residues with similar properties is less likely to break the protein – lead to the creation of amino acid scoring matrices (such as BLOSUM (Henikoff and Henikoff, 1992) and PAM (Dayhoff et al., 1978)) that indicate how favourable is any given change of residues (with no change being the most favourable, in particular for more peculiar residues). Finding an optimal alignment is a computationally complex task – for two sequences with lengths $M$ and $N$ it has the time complexity of $O(N \times M)$ and space complexity $O(N \times M)$ (can be reduced to $O(N \cdot \max(1, \frac{M}{\log(N)}))$ (Arlazarov et al., 1970; Masek and Paterson, 1980), and $O(\min(N, M))$ (Hirschberg, 1975), respectively).

It is possible to generalise the pairwise sequence alignment to encompass a larger number of sequences, creating a multiple sequence alignment (MSA). However, only the aforementioned dynamic programming alignment algorithms can be scaled to a larger number of sequences, and those are matrix based (of the size corresponding to one sequence in each dimension). As such calculating them for a real life biological data set is too time-consuming to globally optimise (as it would require creating and finding an optimal path through an $N$ dimensional matrix of size $L$, where $N$ is the number of sequences and $L$ is the length of a sequence, assuming it is constant for all).

An MSA in turn can be used to create a sequence profile which summarises the diversity of each column, e.g. through a Hidden Markov Model – linear state machine, in which there is a match node for every column of the alignment. HMM trained on a given MSA (e.g. using the Baum-Welch algorithm) describes the expected pattern behind the known sequences, instead of just the observed counts (as is the case for position weight matrices). As such, HMMs are significantly more sensitive for more remote homology detection (Madera and Gough, 2002).
MSAs are commonly calculated using heuristic methods, for example by creating a guide tree based on pairwise distances between sequences, with each node merging the sub-alignments created in its children (Sievers and Higgins, 2014) (see Fig. 5).

An alternative method of building an MSA is by maximising the agreement with all-vs-all pairwise (which can be resolved optimally in polynomial time) alignments. This can be done by finding a maximum weight trace (Kececioglu, 1993) in a graph $G = (V, E, \prec)$ representing a set of alignments, which is defined as follows.

**Definition 2.** A graph $G = (V, E, \prec)$ is an alignment graph for a set $S$ of sequences if vertices $V$ correspond to positions in sequences in $S$, with an order within each sequence $S_i$ imposed by relation $\prec$ on positions $s_i, s_j \in S_i$: $s_i \prec s_j \iff i + 1 = j$, that is $\prec$ holds only if $s_i$ immediately precedes $s_j$. Edges $E$ are undirected weighted connections between vertices which have been aligned in one of the pairwise alignments.

A path in graph $G$ is a set of positions to be aligned in one column – thus splitting the graph into connected components gives columns of the alignment, with the caveat that the alignment is only valid if columns can be linearly ordered under relation $\prec'$, where for connected components $A$ and $B$:

$$A \prec' B \iff (\exists a \in A)(\exists b \in B) : x \prec y.$$

A trace in the alignment graph $G$ is then a subset of edges $T \subseteq E$ for which connected components are acyclic under $\prec'$. For a graph $G$ with edges weighted by function $w$ the maximum weight trace is found by maximising $\sum_{e \in T} w(e)$.

In Chapter 3 we propose two new heuristic algorithms for finding the maximum weight trace (and the resulting multiple alignment), both building the columns of the alignment (connected components in a graph) using a modified Dijkstra’s shortest path tree algorithm. The first in based on greedily extracting columns one by one (a ‘depth-first’ approach), the second, on a bottom-up clustering of vertices till the lowest number of acceptable components is reached (a ‘breadth-first’ approach). Creating an MSA this way has an important advantage compare with other MSA algorithms – restrictions of the input ‘sequence’ are significantly loosened, and in fact sequence profiles can be used instead of plain sequences. We take advantage of that, and present the first phylogenetic analysis of the evolution of slipknotted (that is with a knot present on only a fragment of a chain) proteins.

### Rysunek 5: Creation of a multiple sequence alignment (MSA) using a guide tree.
Multiple sequence alignments show us the present-day sequences (as the only data available), but their diversity can give us some insights into the evolutionary history of e.g. a protein family. In particular, the residues in a protein don’t work in a vacuum – they interact with each other, and mutating one of such interaction partners may influence the evolution of the other. This process is the basis for use of co-evolutionary methods, such as Direct Coupling Analysis (Weigt et al., 2009; Morcos et al., 2011), to study the relationship between protein sequences and structures. For a multiple sequence alignment, DCA calculates a fully-connected statistical model of residue probabilities, which gives direct (not through another residue) correlation scores, called Direct Information (DI), to all pairs of positions in the MSA. It is a statistical inference framework built on a Potts model, which represents \( q \) types of interacting spins on a lattice model. Fitting this to an MSA of length \( N \) gives a model with an \( N \times N \) lattice with \( q = 21 \) spins (representing the symbols in the alignment – 20 standard amino acids and a gap). For the alignment as a whole, each spin on the lattice is in fact better presented as a \( q \times q \) matrix of coincidences between corresponding columns in the alignment. Under this model we can calculate the probability of a sequence to match the model, as well as parameters representing the strength of direct (without the input from the rest of the sequence) interactions between any two positions.

Second part of Chapter 3 goes into more detail on applications of DCA to protein research. We present DCA-MOL, a tool for easy exploration of Direct Information in regards to a known structure (Jarmolinska et al., 2019b). Then we describe PConsFam, a database of protein structures modeled based on DCA scores (Lamb et al., 2019). Finally, we show how introduction of DCA-found interactions can facilitate protein folding simulations (Dabrowski-Tumanski et al., 2015).

**Databases and algorithmic tools for protein topology explorations**

Chapter 4 describes our other works in the field of computational protein science. It was recently shown (Dabrowski-Tumanski and Sulkowska, 2017) that topological non-triviality in proteins need not be confined in one chain – examples of multichain links have been found. We present an online, self-updating database which gathers in one place information about linked protein chains in all published structures – LinkProt (Dabrowski-Tumanski et al., 2016).

Even with all the advances in protein structure determination techniques, such as crystallography and electron microscopy, some structures still cannot be resolved in full. While for some applications this is a mere annoyance, such 'broken' – gapped – structures cannot be used e.g. in molecular dynamics simulations. There are tools for modeling gaps, however they are often limited in scope, and some even actively unknot any non-trivialities in final structure. To allow for topologically aware filling of missing fragments in structures, we
created GapRepairer (Jarmolinska et al., 2018), a gap-modeling web server.

Finally, using coarse-grained molecular dynamics simulations, we propose folding pathways of several newly discovered knotted proteins (Jarmolinska et al., 2019a).
Publications included in Chapter 2


Publications included in Chapter 3


Publications included in Chapter 4


Other Publications


