# VU Research Portal

## Structure Alignment

Ivanova, Olga; Gavaldá-Garciá, Jose; Gogishvili, Dea; Houtkamp, Isabel; Bouwmeester, Robbin; Feenstra, K. Anton; Abeln, Sanne

**Link to publication in VU Research Portal**

# Chapter 3

# Structure Alignment

**Olga Ivanova** iD    **Jose Gavaldá-García** iD    **Dea Gogishvili** iD
**Isabel Houtkamp** iD    **Robbin Bouwmeester** iD
**K. Anton Feenstra*** iD    **Sanne Abeln*** iD
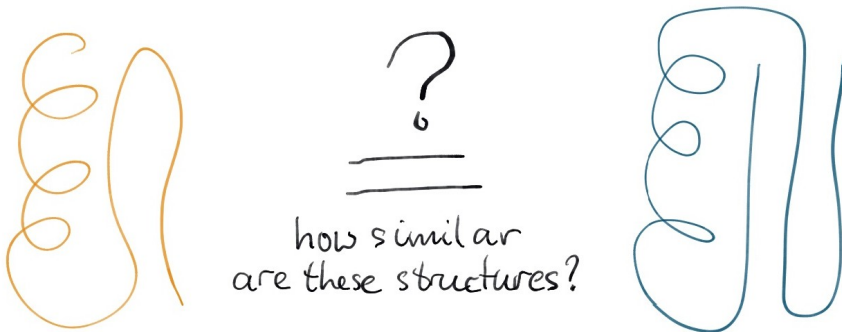
* editorial responsability

# 3 Structure Alignment 1

Olga Ivanova ⓘD    Jose Gavaldá-García ⓘD    Dea Gogishvili ⓘD

Isabel Houtkamp ⓘD    Robbin Bouwmeester ⓘD

K. Anton Feenstra* ⓘD    Sanne Abeln* ⓘD

# 1   Comparing protein structures

The Protein DataBank (PDB) contains a wealth of structural information (Berman *et al.*, 2000). In order to investigate the similarity between different proteins in this database, one can compare the primary sequence through pairwise alignment and calculate the sequence identity (or similarity) over the two sequences. This strategy will work particularly well if the proteins you want to compare are close homologs. Knowledge of sequence similarity is widely used in for example the Pfam database (Finn *et al.* 2014, see Chapter "Data Resources for Structural Bioinformatics"): Pfam cluster similar proteins into families based on the sequence of the protein. However, in this chapter we will explain that a structural comparison through structural alignment will give you much more valuable information, that allows you to investigate similarities between proteins that cannot be discovered by comparing the sequences alone.

Furthermore, we will discuss the challenges in understanding how similar two structures are based on **structural information alone** (i.e. the atomic coordinates, see also Figure 3.1); this means we do not use any sequence information on the two proteins. Using solely the structure for comparison poses a difficult computational problem due to the many possible ways to align the structure. However, structural comparison is generally considered to be more reliable and evolutionary accurate than a comparison based on sequence similarity.
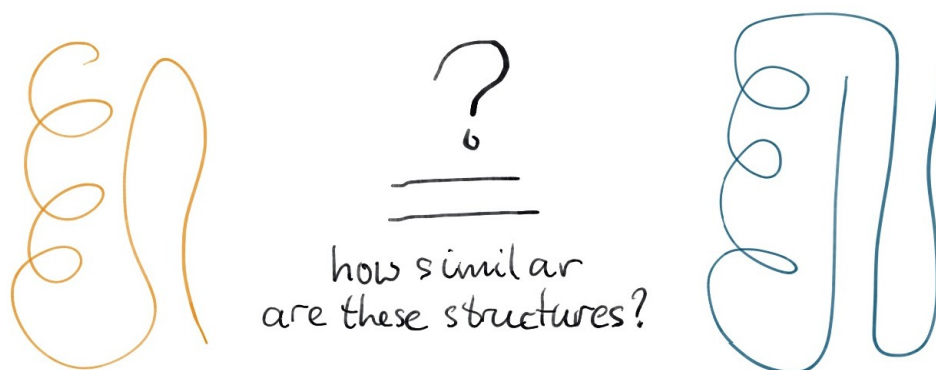


*Figure 3.1: Structural alignment deals with the problem of determining how similar two structures are – based on the atomic coordinates alone (no sequence information).*

## 1.1   Structure is more conserved than sequence

One of the reasons why structural comparison is valuable, is that structure is generally more conserved than sequence. In evolution, selection pressure
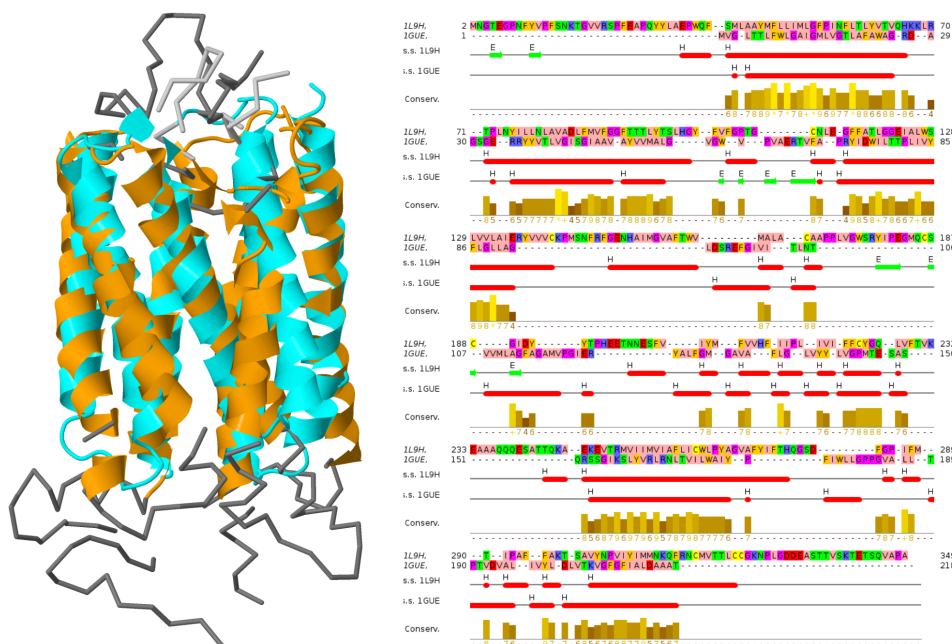
Figure 3.2: Structure is more conserved than sequence. Left: The output of a structural alignment program, Combinatorial Extension (CE). These two proteins (orange and cyan) have, as you see, a similar structure. They are both rhodopsins, and they have a similar function (light detection). However, their sequence identity (right) is **less than 5%**. This is below the similarity you would expect from two random sequences. Note that one would not be able to align these proteins using sequence identity alone. One can see that the positioning of the helices are very well conserved between the two structures, but that there is much more variability in the loops (both in structure and in length). The two proteins are bovine rhodopsin (PDB:1L9H, in orange) and sensory rhodopsin (PDB:1GUE, in cyan). Website at http://www.rcsb.org/pdb/workbench/workbench.do

acts on the function of a protein. If unfolded, the majority of proteins will lose their function. Moreover, it may be dangerous for the cell if a protein is partially folded or misfolded because hydrophobic residues would be exposed leading to the protein aggregation. This effectively puts an evolutionary pressure on the ability of a protein to fold into a functional state and on the folding process itself (Tartaglia and Vendruscolo, 2009).

Structure being more conserved than sequence has several important consequences for methods in structural bioinformatics. This importance is for example illustrated by the large number of structure prediction methods that use these principles as a foundation, and also by the design principles of structural classification databases. Importantly, distant evolutionary relations between two proteins are more easily observed by comparing structures rather than sequences.

An example of structure conservation without sequence conservation is

demonstrated in Figure 3.2, which shows the typical output of a structural alignment program. Two homologous proteins, i.e., with common ancestor, are aligned with the structural alignment program Combinatorial Extension (CE, by Shindyalov and Bourne, 1998, on the left) and superimposed on top of each other (on the right). Note that the sequence identity between the proteins is very low; so low in fact, that two *random* sequences aligned by sequence similarity would give a similar score. However, from the structural comparison in Figure 3.2, it is clear that the structures are much alike. In this case, only the structural alignment allows the correct identification of common ancestry for these two proteins.

# 2    Structural superposition

The simplest manner to compare two different configurations of a protein is by generating a *structural superposition* of the two structures. In this section we will focus on the superposition problem, i.e. how to "overlay" two protein structures in 3D space (see Figure 3.3). Keep in mind that if we want to superimpose two structures, we need a mapping between the residues of each structure ( i.e. an alignment - but not based on the sequence). In the next section, *Structural alignment*, we will see how we can compare structures if we do not have an alignment.

## 2.1    PDB coordinates as a structure representation

In Chapter "Data Resources for Structural Bioinformatics" we will see how a protein structure may be represented and stored in a PDB record in more detail. For now it is enough to know that the structure is recorded in the PDB file by assigning an x, y and z coordinate for each atom (Equation 1). Hence each atom can be represented by a positional vector $\mathbf{p} \in \mathbb{R}^3$ in three dimensional space, such that $\mathbf{p} = (p_x, p_y, p_z)$.

Note that a protein structure can be rotated as a rigid body, without changing the interatomic distances. It is up to the experimentalist who created the PDB file to record what the frame of reference is, as any arbitrary rotation and translation may be given. Comparing the rotation and translation of two different protein structures is a key part of the structural superpositioning. Before we go into any more detail, we will first need a score to evaluate the similarity of two protein structures.

## 2.2    A score for comparing protein structures – RMSD

The root mean square deviation (RMSD), is one of the simplest measures to score the similarity of two protein structures. RMSD calculates the squared difference between two sets of atoms. In practice, a single representative
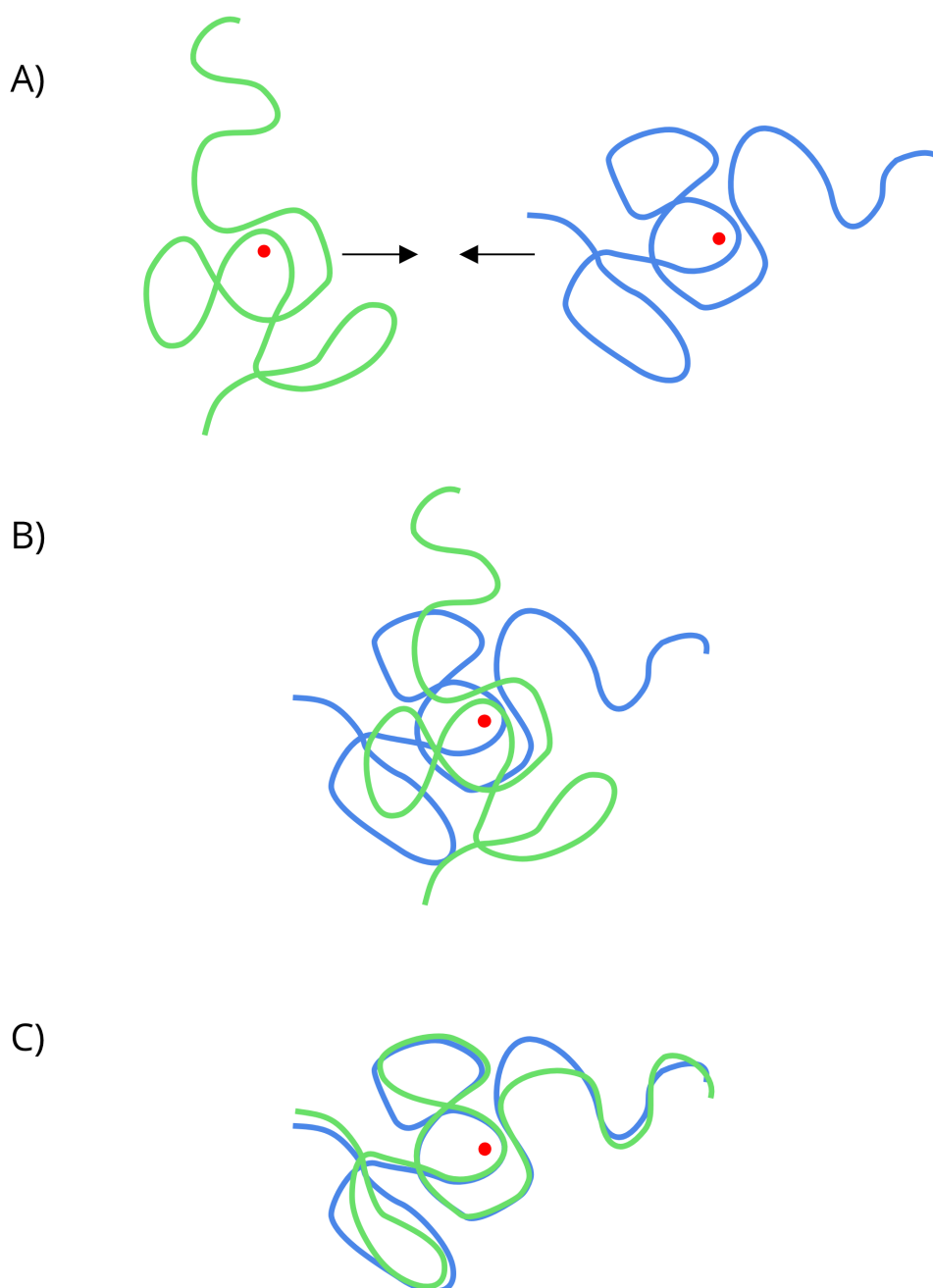
Figure 3.3: **Superimposing two protein structures.** *The lines represent proteins that need to be aligned and the red dots indicate their calculated center of mass. The superposition problem is explained by figures A), where we need to find the "best" overlay in which the the two structures can be compared. C) shows a solution to the problem. The process of superimposing two structures: A) The centers of mass for the proteins are calculated using Equation 2. B) The centers of mass of both proteins are put in the same spatial coordinate. C) The protein structures are superimposed.*

atom per residue is chosen, i.e., C-alpha or C-beta atoms. The RMSD of two structures $V$ and $W$, given the residues mapping $M$, is given by:

$$\begin{aligned} \text{RMSD}(V,W) &= \sqrt{\frac{1}{n} \sum_{i \in M} \|\mathbf{v_i} - \mathbf{w_i}\|^2} \\ &= \sqrt{\frac{1}{n} \sum_{i \in M} (v_{i,x} - w_{i,x})^2 + (v_{i,y} - w_{i,y})^2 + (v_{i,z} - w_{i,z})^2} \end{aligned} \tag{1}$$

Here we calculate the squared distance between atoms $\mathbf{v_i}$ of structure $V$ and atoms $\mathbf{w_i}$ of structure $W$. Note the sum over $i$ runs over all matched pairs $M$ of atoms in the alignment of $V$ and $W$; the total number of matched (aligned) residues is $n$. This means we need to know how the residues correspond in the two different structures in order to calculate this score, i.e. we need to know which $w_i$ we need to subtract from which $v_i$.

## 2.3  Structural superposition and RMSD

The method of **finding an optimal rotation and translation** is called **structural superposition**. An RMSD score between the two sets of atoms is minimised to determine the optimal rotation. Hence, a superposition algorithm provides the rotation that yields the best RMSD fit. Importantly, structural superposition cannot identify the alignment or residues between two protein structures; we will come back to this at the end of this section.

Note that the RMSD score only gives a measure of the dissimilarity between two protein structures, if they have first been superimposed; without superposition, one would merely measure how far apart the structures are in the given coordinate frame(s), as explained in Figure 3.3. More formally, a measure for structural dissimilarity should be described as the 'RMSD after least-squares-fitting', instead of simply using 'RMSD', but this is typically omitted in research papers (and even most text books).

Figure 3.3 also illustrates how one of the two structures needs to be translated and rotated in order to obtain a minimal RMSD between corresponding atom sets. The optimal translation can be found by simply ensuring that the two centers of mass will fall on top of each other. The coordinates for the center of mass of each structure, $\mathbf{R} \in \mathbb{R}^3$, may be found using Equation 2.

$$\mathbf{R} = \frac{\sum_{i=0}^{i=N} m_i \mathbf{r_i}}{\sum_{i=0}^{i=N} m_i} \tag{2}$$

This is the mass-weighted average of the atomic positions $\mathbf{r_i}$; here $N$ is the total number of atoms, and $m_i$ is the mass of each atom $i$. Figure 3.4 exemplifies the meaning of the center of mass.

Since we are typically using a single type of atoms (i.e. C-alpha), the masses ($m_i$) in Equation 2 cancel out. Now we can calculate the directional
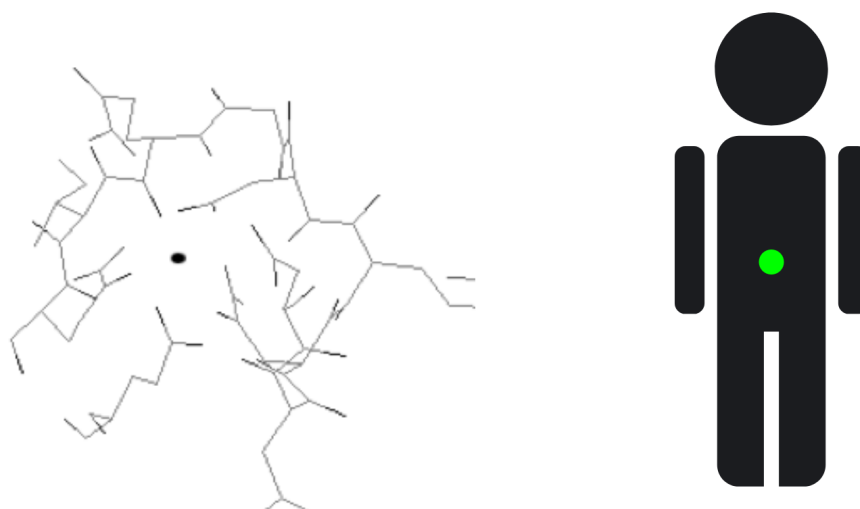
Figure 3.4: **Center of mass in molecules and humans.** *The center of mass is the spatial position determined by the average of all the atoms in a system. The center of mass of a molecule (left, black dot) is calculated the same way that a center of mass for a human (right, green dot) would be calculated. Without going into further detail, the center of mass is placed in the same location as the center of gravity for most objects on Earth, including our bodies.*

difference between the two centers of mass to calculate the required translation. Note that a translation in three-dimensional space means that one needs to add or subtract an identical vector, in this case corresponding to the center of mass, from the coordinates of all atoms.

Once the centers of masses have been superimposed, there exists a single rotation that will minimize the RMSD. This rotation can be found exactly; in practice, you need to transform three-dimensional coordinates, to 4D quaternions (Kearsley, 1989), and solve an eigenvalue problem using the Jacobi algorithm. We will not go through this method in detail, as it provides little extra understanding of the practical problem. Computationally, it is not very expensive to solve this problem: the exact solution can be found in polynomial time.

Structural super-positioning can be used to compare protein structures for which the mapping of residues is already known. For example, if we want to compare snapshots in a simulation to the native protein structure, we want to actually compare a protein with itself; hence the mapping of the residues is trivial. Similarly, if we want to compare a model from a structure prediction method, with the experimentally determined structure, we would also use structural super-positioning directly. However, if we want to compare (potentially) homologous proteins, the mapping of residues may not be known, and we need structural alignment to generate such a mapping.

# Structural superposition   Structural alignment
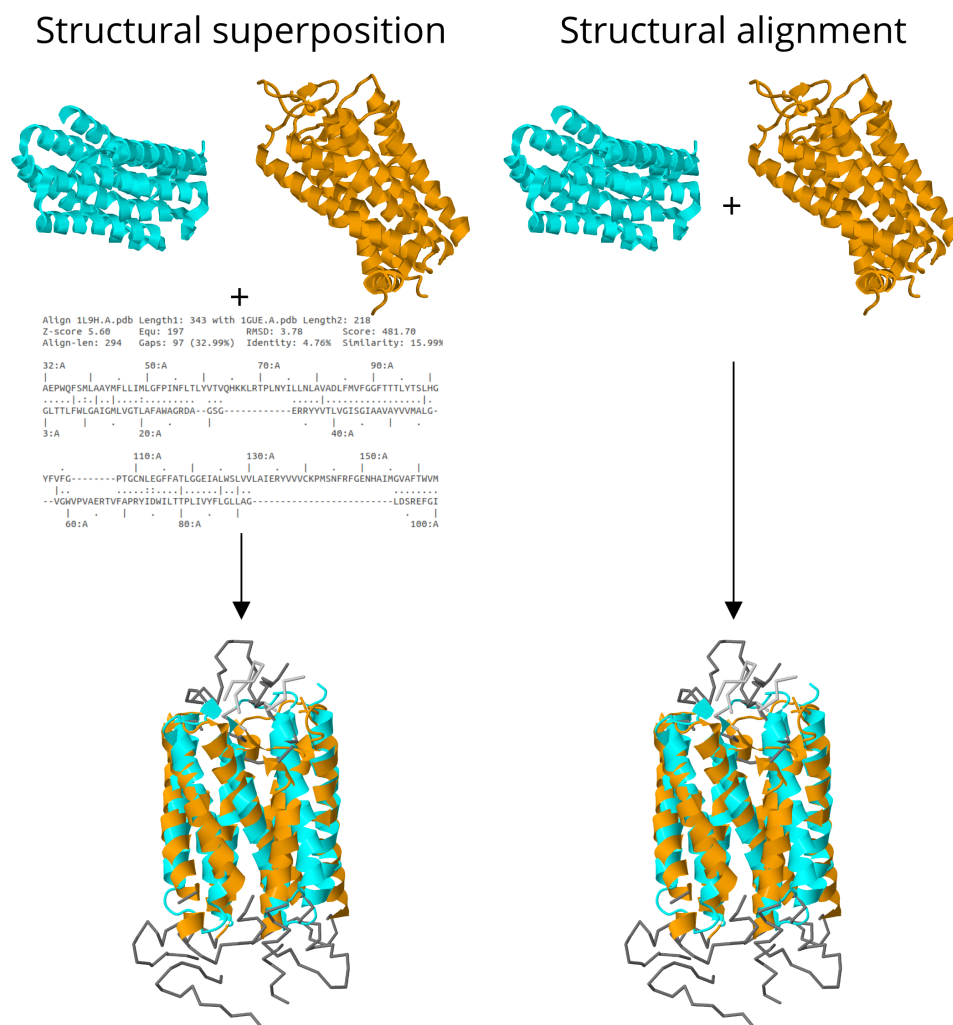


Figure 3.5: **Structural superposition versus structural alignment.** *Left: Structural superposition requires the structures of the proteins and an alignment of the residues as input. Note that if the two structures originate from one protein (and thus have the same sequence) the alignment of the residues is trivial. The superposition method works by minimizing the RMSD, for which we need a mapping (alignment) between the residues. The structural superposition will return two structures in the same frame of reference, such that the RMSD may be calculated. Right: Structural alignment takes the protein structures as its only input. The method will try to match similar substructures between the proteins. It will return an alignment, as well as a score for the (dis)similarity of two protein structures.*

## 3   Structural Alignment

Structural alignment deals with the problem of generating a mapping between residues, or an alignment, based solely on the structure of two pro-

teins. Note that this mapping of residues is not a trivial task. One may be tempted here to use the sequence as a reference. However, as illustrated by Figure 3.2, this would be a poor choice, as the sequence similarity may be very low for similar structures. Hence we would like to use structure alone to map the corresponding residues (e.g. C-alpha or C-beta atoms) of two structures to be superimposed. The problem of **finding the optimal match or alignment between the residues of two protein structures** is called **structural alignment**.

A typical output of a structural alignment method will therefore provide an alignment of the residues, based on the structure of the backbone alone. Typically, also a superposition of the two structures will be given as an output, as well as a structural alignment score. Please also consider Figure 3.5 for an explanation of the difference between structural alignment and structural superposition.

The problem of structural alignment is computationally much more difficult to solve than the superposition problem. In fact, it has been suggested to belong to a class of computational problems that is called NP-hard (Hasegawa and Holm, 2009). In practice, this means that the time required for the computation grows exponentially with the size of the input (protein lengths in this case). A consequence is that for many real sized proteins, of about 200-300 residues, it would be very difficult to find the optimal solution, since it is computationally too expensive to search through all possible structural alignments. However, some methods can provide exact solutions for real size proteins (Wohlers *et al.*, 2012).

### Computational complexity of structural alignment versus sequence alignment

Note that the pairwise structural alignment problem is very different in terms of computational complexity from the pairwise sequence alignment problem. For the latter problem we can find an optimal solution, by filling in the dynamic programming matrix. Using dynamic programming, the pairwise sequence alignment problem can be solved exactly in $O(n \cdot m)$ time for aligning one sequence of length $n$ with one of length $m$, since $n \times m$ operations need to be calculated to fill in the dynamic programming matrix.

The difference between these two alignment problems is that there are no natural local bounds in structural alignment. Imagine that we have an already aligned region of the sequence. For sequence alignment, if we change the alignment outside this region, no change of scoring will take place inside the already aligned region. However, with structural alignment, the introduction of a gap outside the aligned re-

gion may also affect the score of the already aligned region, due to
residues outside the already aligned region that are close in 3D space
(but not in sequence space).

## 3.1   The three key components of structural alignment

Structural alignment methods all contain three crucial parts: a suitable
*structural representation* for the proteins, a method to *optimize* a similarity
measure and a (statistical) *score for protein structure similarity* (Marti-
Renom *et al.*, 2009). Plenty of different representations, methods and mea-
sures have been used to tackle the problem of structural alignment; here, we
will give a brief overview of some of the most important strategies.

## 3.2   Structure representation and contact maps

In order to find a good structural alignment, methods typically encode the
structure of both proteins into a representation that can be more easily
compared between residues of the two proteins. Note that it is important
that this representation is invariant to the frame of reference, otherwise it
difficult to compare substructures; hence, the c-alpha coordinates are not
suitable.

An example of a suitable representation is the *contact map*, or contact
matrix. Note that such a contact map is defined for a single protein struc-
ture, not for a pair of structures.

A contact can be defined as two residues that are close in three-dimensional
space. We can now create a contact map, by considering for each pair of
residues in a protein if they make a contact, see for example Figure 3.6.

A contact matrix, $C_{i,j}$, can be mathematically formalized as follows.

$$C_{i,j} = \begin{cases} 1 & \text{if i and j make contact} \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Here all possible residues pairs $(i,j)$ are considered in a single protein struc-
ture. A cutoff of around 7 Åis typically used to define which residues are in
contact.

From such a map, secondary structure elements may be recognized: for
example, alpha-helices can be identified as a diagonal line adjacent to the
diagonal. This is because in a helix there is a typical hydrogen bonding
pattern of residue $i$ and $i + 4$ along the sequence, see also Figure 3.6.

We can compare the local three-dimensional surroundings of a residue
by considering the contacts. If residues in two different protein structures
have (locally) similar contact maps, they have similar substructures. Using
contact maps, the structural alignment problem can be redefined as the
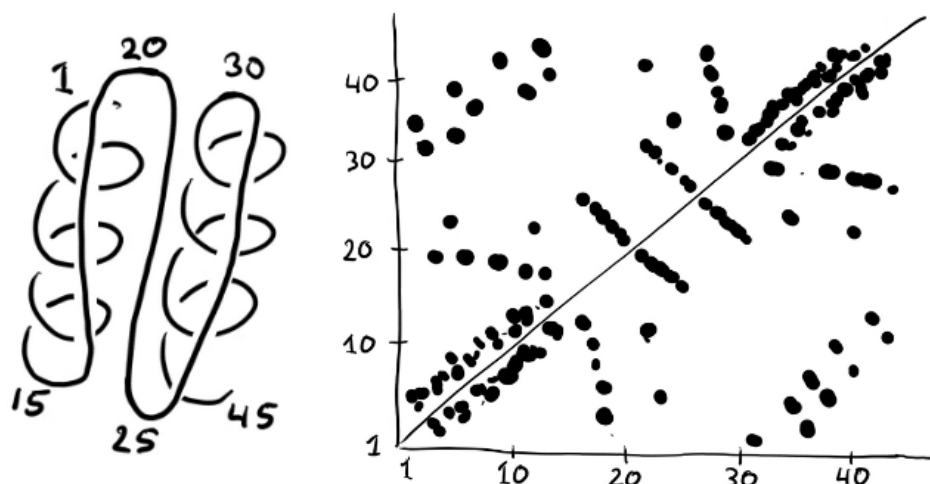problem to find an optimal alignment between two contact maps.

Figure 3.6: **Contact map of a single protein.** *Here it can be observed which atoms are "in contact" (closer than a set distance) in a protein structure. From this figure it can be easily observed the the alpha helices close to the main diagonal, and the interaction between residues due to the tertiary structure.*

Similarly, a *distance map* or distance matrix may be defined for a protein structure. In this case the matrix elements represent the distance between residues, instead of the boolean value indicating if there is a contact or not. The structural alignment method Dali (Holm and Laakso, 2016), which is still considered to be highly accurate to this day, uses distance matrices to represent structures Other examples of structure representation are c-beta vectors as used in SSAP(Taylor and Orengo, 1989; Orengo and Taylor, 1996) and URMSD vectors as used by MAMMOTH (Lupyan *et al.*, 2005).

## 3.3 Heuristic optimization algorithms

Once we have a good structural representation in which two structures, or the substructures thereof, can be compared, we can formulate structural alignment as an optimisation problem and search for its solution. In the case of the contact matrix, we would look for an alignment of residues in which the corresponding residues have the most similar contact patterns. Since, searching for this optimal solution is thought to be an NP-hard problem (see previous section), performing an exhaustive search for the optimization problem is computationally not feasible. Instead, one needs to use a heuristic search strategy. Heursitic methods search for a good alignment, but cannot guarantee that the solution is optimal. Many of such heuristic search algorithms exist. Examples used for structural alignment are the branch and bound algorithm (Wohlers *et al.*, 2012) and Monte Carlo optimization.

In addition, double dynamic programming (Orengo and Taylor, 1996) and (single) dynamic programming (van Kempen *et al.*, 2023) are also used as optimisation strategies.

## 3.4   Statistical scoring of structural alignments

Next to the actual alignment, a structural alignment method will also provide a (dis)similarity measure for the two structures as an output. This is typically based on the alignment score, and is closely related to the used structure representation.

We described previously how RMSD can be used to score a structural superposition. Once a structural alignment is made, we could use RMSD as a measure to indicate how similar two proteins are. Most structural alignment programs will provide the RMSD as an output value for the superposition of the best structural alignment found.

However, there are some caveats with using RMSD as a measure for structural similarity. Note that, RMSD between two protein structures depends on the size of the proteins being compared. For two random structures, the average distance will become larger if the proteins being compared are large (e.g. Maiorov and Crippen, 1995). A similar effect may occur when one of the proteins that is being compared has many residues far away from the center of mass, for example if it has long loops. Hence, RMSD can be very sensitive to these 'outlier' atoms or residues in the structure.

Apart from the RMSD, structural alignment methods will typically also calculate a p-value or z-score, indicating how significant the structural alignment is. This is important, as the raw comparison scores (e.g. RMSD or overlap in contact maps) are intrinsically length dependent. If we want to see if two structures are significantly similar, this needs to be taken into account: two large structures only sharing two secondary structure elements are typically not homologous.

One can estimate distributions of raw alignment scores over a set of structural alignments between random (non-homologous) structures, for different sizes of structures. From these distributions, subsequent z-scores or p-values may be derived. Note that sequence alignment methods follow a similar strategy to derive statistical scores. For example, BLAST (Altschul *et al.*, 1997) uses statistical e-values in which database size, and sequence length are accounted for.

# 4   Applications of structure comparison

Applications of structural comparison and alignment are diverse but mostly concern the fields of evolutionary and structural biology. The majority of structural comparison applications are used to detect remote homology. Entire protein classification databases such as SCOP (Andreeva *et al.*, 2008),

CATH (Dawson *et al.*, 2017) and also the PDB (Berman *et al.*, 2000) use structural alignment methods to find and cluster structurally similar proteins. With increasing database sizes, and the availability of a larger number available predicted structures (Varadi *et al.*, 2022), the speed of these methods becomes increasingly important (van Kempen *et al.*, 2023). In Chapter "Data Resources for Structural Bioinformatics" we will look at some of these resources in more detail.

The identification of distant homology with structure alignment also allows the prediction of protein's functions, (Roy *et al.*, 2012) and the classification of proteins with known structures. Structural comparison and alignment ease the prediction of protein structure from the sequence, (Ma *et al.*, 2012, 2013) and enable the identification of structure patterns and the observation of protein folding space (Yang and Honig, 2000; Kolodny *et al.*, 2006). A less known but significant application of structural comparison and alignment is the recognition of binding sites (Wass *et al.*, 2010).

# 5 Key points

- Structural superposition can be used to calculate the RMSD or another similarity metric between two structures, and requires a mapping between residues to do so.
- Structural alignment is used to determine a mapping between residues based on structure alone.
- Structure provides better insight into evolutionary processes because structure is more conserved than sequence.
- Contact maps can be used to represent a protein structure and its substructures in a rotation invariant fashion.
- Heuristic methods for score optimization are employed in structural alignment to make computational costs feasible.

# 6 Further reading

For a full overview please see Structural Alignment Chapter by Marti-Renom *et al.* (2009) in the book "Structural Bioinformatics" by Gu and Bourne (2009).

# Author contributions

| | |
|---|---|
| Wrote the text: | OI, DG, SA, KAF |
| Created figures: | JG, SA, KAF |
| Review of current literature: | OI, RB, SA |
| Critical proofreading: | BS, SA, IH |
| Non-expert feedback: | HM, TL, RB |
| Editorial responsibility: | KAF, SA |

The authors thank Halima Mouhib [iD], Robbin Bouwmeester Robbin Bouwmeester, and Ting Liu [iD] for critical proofreading.

# References

Altschul, S. F., Madden, T. L., Schäffer, A. A., Zhang, J., Zhang, Z., Miller, W., and Lipman, D. J. (1997). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic acids research*, **25**(17), 3389–402.

Andreeva, A., Howorth, D., Chandonia, J. M., Brenner, S. E., Hubbard, T. J. P., Chothia, C., and Murzin, A. G. (2008). Data growth and its impact on the SCOP database: New developments. *Nucleic Acids Research*, **36**(SUPPL. 1), 419–425.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. (2000). The Protein Data Bank. *Nucleic Acids Res*, **28**(1), 235–242.

Dawson, N. L., Lewis, T. E., Das, S., Lees, J. G., Lee, D., Ashford, P., Orengo, C. A., and Sillitoe, I. (2017). CATH: An expanded resource to predict protein function through structure and sequence. *Nucleic Acids Research*, **45**(D1), D289–D295.

Finn, R. D., Bateman, A., Clements, J., Coggill, P., Eberhardt, R. Y., Eddy, S. R., Heger, A., Hetherington, K., Holm, L., Mistry, J., Sonnhammer, E. L. L., Tate, J., and Punta, M. (2014). Pfam: the protein families database. *Nucleic acids research*, **42**(Database issue), 222–30.

Gu, J. and Bourne, P. E. (2009). *Structural bioinformatics*. John Wiley & Sons,, Hoboken, 2nd ed. nv edition.

Hasegawa, H. and Holm, L. (2009). Advances and pitfalls of protein structural alignment. *Current Opinion in Structural Biology*, **19**(3), 341–348.

Holm, L. and Laakso, L. M. (2016). Dali server update. *Nucleic Acids Research*, **44**(W1), W351–W355.

Kearsley, S. K. (1989). On the orthogonal transformation used for structural comparisons. *Acta Crystallographica Section A*, **45**(2), 208–210.

Kolodny, R., Petrey, D., and Honig, B. (2006). Protein structure comparison: implications for the nature of 'fold space', and structure and function prediction. *Current Opinion in Structural Biology*, **16**(3), 393–398.

Lupyan, D., Leo-Macias, A., and Ortiz, A. R. (2005). A new progressive-iterative algorithm for multiple structure alignment. *Bioinformatics*, **21**(15), 3255–3263.

Ma, J., Peng, J., Wang, S., and Xu, J. (2012). A conditional neural fields model for protein threading. *Bioinformatics*, **28**(12), 59–66.

Ma, J., Wang, S., Zhao, F., and Xu, J. (2013). Protein threading using context-specific alignment potential. *Bioinformatics*, **29**(13), 257–265.

Maiorov, V. N. and Crippen, G. M. (1995). Size-independent comparison of protein three-dimensional structures. *Proteins: Structure, Function, and Genetics*, **22**(3), 273–283.

Marti-Renom, M. A., Capriotti, E., Shindyalov, I. N., and Bourne, P. E. (2009). Structure Comparison and Alignment. In J. Gu and P. E. Bourne, editors, *Structural Bioinformatics, 2nd Edition*, pages 397–418. John Wiley & Sons, Inc.

Orengo, C. A. and Taylor, W. R. (1996). [36] SSAP: Sequential structure alignment program for protein structure comparison. In *Methods in Enzymology*, volume 266, pages 617–635. Academic Press.

Roy, A., Yang, J., and Zhang, Y. (2012). COFACTOR: An accurate comparative algorithm for structure-based protein function annotation. *Nucleic Acids Research*, **40**(W1), 471–477.

Shindyalov, I. N. and Bourne, P. E. (1998). Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Engineering Design and Selection*, **11**(9), 739–747.

Tartaglia, G. G. and Vendruscolo, M. (2009). Correlation between mRNA expression levels and protein aggregation propensities in subcellular localisations. *Molecular BioSystems*, **5**(12), 1873–1876.

Taylor, W. R. and Orengo, C. A. (1989). Protein structure alignment. *Journal of Molecular Biology*, **208**(1), 1–22.

van Kempen, M., Kim, S. S., Tumescheit, C., Mirdita, M., Lee, J., Gilchrist, C. L. M., Söding, J., and Steinegger, M. (2023). Fast and accurate protein structure search with Foldseek. *Nature Biotechnology*.

Varadi, M., Anyango, S., Deshpande, M., Nair, S., Natassia, C., Yordanova, G., Yuan, D., Stroe, O., Wood, G., Laydon, A., Zídek, A., Green, T., Tunyasuvunakool, K., Petersen, S., Jumper, J., Clancy, E., Green, R., Vora, A., Lutfi, M., Figurnov, M., Cowie, A., Hobbs, N., Kohli, P., Kleywegt, G., Birney, E., Hassabis, D., and Velankar, S. (2022). AlphaFold Protein Structure Database: massively expanding the structural coverage of protein-sequence space with high-accuracy models. *Nucleic Acids Research*, **50**(D1), D439–D444.

Wass, M. N., Kelley, L. A., and Sternberg, M. J. E. (2010). 3DLigandSite: Predicting ligand-binding sites using similar structures. *Nucleic Acids Research*, **38**(SUPPL. 2), 469–473.

Wohlers, I., Malod-Dognin, N., Andonov, R., and Klau, G. W. (2012). CSA: Comprehensive comparison of pairwise protein structure alignments. *Nucleic Acids Research*, **40**(W1), 303–309.

Yang, A. S. and Honig, B. (2000). An integrated approach to the analysis and modeling of protein sequences and structures. I. Protein structural alignment and a quantitative measure for protein structural distance. *Journal of Molecular Biology*, **301**(3), 665–678.