The Impact of Dynamics in Protein Assembly

LUCAS SEBASTIAN POWELL-RUDDEN

The Impact of Dynamics in Protein Assembly

Lucas S.P. Rudden

A thesis presented for the degree of Doctor of Philosophy

Department of Physics

Durham University
UK

February 2021

The Impact of Dynamics in Protein Assembly

Lucas S.P. Rudden

Submitted for the degree of Doctor of Philosophy
February 2021

Abstract

Predicting the assembly of multiple proteins into specific complexes is critical to understanding their biological function in an organism, and thus the design of drugs to address their malfunction. Consequently, a significant body of research and development focuses on methods for elucidating protein quaternary structure. *In silico* techniques are used to propose models that decode experimental data, and independently as a structure prediction tool. These computational methods often consider proteins as rigid structures, yet proteins are inherently flexible molecules, with both local side-chain motion and larger conformational dynamics governing their behaviour. This treatment is particularly problematic for any protein docking engine, where even a simple rearrangement of the side-chain and backbone atoms at the interface of binding partners complicates the successful determination of the correct docked pose. Herein, we present a means of representing protein surface, electrostatics and local dynamics within a single volumetric descriptor, before applying it to a series of physical and biophysical problems to validate it as representative of a protein. We leverage this representation in a protein-protein docking context and demonstrate that its application bypasses the need to compensate for, and predict, specific side-chain packing at the interface of binding partners for both water-soluble and lipid-soluble protein complexes. We find little detriment in the quality of returned predictions with increased flexibility, placing our protein docking approach as highly competitive *versus* comparative methods. We then explore the role of larger, conformational dynamics in protein quaternary structure prediction, by exploiting large-scale Molecular Dynamics simulations of the SARS-CoV-2 spike glycoprotein to elucidate possible high-order spike-ACE2 oligomeric states. Our results indicate a possible novel path to therapeutics following the COVID-19 pandemic. Overall, we find that the structure of a protein alone is inadequate in understanding its function through its possible binding modes. Therefore, we must also consider the impact of dynamics in protein assembly.

Declaration

The work in this thesis is based on research conducted within the Degiacomi research group at Durham University, UK; initially from October 2017 at the Department of Chemistry, and, from October 2019 onward, the Department of Physics. No part of this thesis has been submitted for consideration elsewhere for any other degree or qualification. Everything written here is my own work unless indicated otherwise in the text.

Acknowledgements

To all the friends, family, teachers, colleagues and acquaintances who have carried me on my long journey through the entirety of the UK's education system, thank you. Most especially to my mum, from reading my history essays to waking me up in the morning with instant coffee. I couldn't have done this without your unwavering support. To my Chemistry teacher Mr Payne at CCF, for inspiring me, and for all the important lessons that weren't about Chemistry. To Chloe, who has been my rock throughout my time at Durham, and who I've tortured with every single piece of written work during my PhD, and in the times before. To all of my undergraduate friends who followed me on this path: Dan, Colin, Vanessa and Jack, and to the new friends of CG200X past and present: Martin (ruler of MartinLand), Tom, Gary, Sarah, Andrew, Golda, Vanessa and Lewis, who brought joy and laughter to my time in the office.

Thank you to the collaborators who made much of the work in this thesis possible. To Chris, for teaching me pretty much everything you see in Chapter 4. To our 2019/20 Masters student George, who did some preliminary work for Sections 2.6 and 4.3. To Valentina, who guided me through my Masters, and for introducing me to my supervisor. And so, to Matteo, who is, without a doubt, the greatest supervisor I could have possibly asked for. I am privileged, honoured, and extraordinarily lucky that you chose me as your first PhD Student, in what I'm confident will be a hugely successful academic career. From your resolve to always pursue loose ends and make sure we're correct (all hail the true King of Benchmarks), to staying humble and never being afraid to admit a mistake. Your kindness – both your continuing support and friendship, to everyday things like helping that lady on the train back from Sheffield. Your creativity; thinking up novel solutions to challenging problems, and interesting projects we could pursue. Your open-mindedness and willingness to listen to ideas, a trait my friends tell me are unusual in academic supervisors. These are the hallmarks of not just a great teacher, but of a great person. You will forever be one of the great teaching influencers in my life, and I am envious of anyone that gets the chance to be a part of your group. One day, I hope to make you proud, as you make me proud everyday with your continued dedication to Science and life in general. I just hope I don't spend too long analysing that trampled flower on my journey.

Contents

Publications & Manuscripts

Section 2.2, 2.3, 3.2 and 3.3 contain modified extracts pertaining to the creation and benchmarking of STID maps, as well as its application to docking.
Lucas S. P. Rudden *and Matteo T. Degiacomi*
Protein Docking Using a Single Representation for Protein Surface, Electrostatics and Local Dynamics
Journal of Chemical Theory and Computation, 2019, 15(9), 5135-5143, DOI: `10.1021/acs.jctc.9b00474`

Section 2.5 focuses on how STID maps can inform on the relationship between the density of a protein and its local dynamics.
Lucas S. P. Rudden*, Erik G. Marklund, Matteo T. Degiacomi*
On the Relationship between Protein Density and Dynamics
In preparation

Section 3.4 shows how our protein-protein docking software released with the 2019 paper, can also be extended to work with transmembrane protein complexes.
Lucas S. P. Rudden*, Matteo T. Degiacomi*
Transmembrane protein docking with JabberDock
Journal of Chemical Information and Modeling, 2021, 61(3), 1493-1499, DOI:`10.1021/acs.jcim.0c01315`

Section 3.5.2 discusses our contribution to a collaborative publication attempting to qualify the utility of mass photometry for integral membrane protein structure characterisation.
Anna Olerinyova, Adar Sonn-Segev, Joseph Gault, Cédric Eichmann, Johannes Schimpf, Adrian H. Kopf, **Lucas S. P. Rudden***, Dzmitry Ashkinadze, Radoslaw Bomba, Jason Greenwald, Matteo T. Degiacomi, J. Antoinette Killian, Thorsten Friedrich, Roland Riek, Weston B. Struwe, Philipp Kukura*
Mass Photometry of Membrane Proteins
Chem, 2020, 7, 1-13, DOI: `10.1016/j.chempr.2020.11.011`

Section 4.5 is a modifed extract from a collaborative conference paper on a new algorithm that ray marches 3D non-convex shapes for efficient collision detection.
Adam Leach, ***Lucas S. P. Rudden****, Sam Bond-Taylor, John C. Birgham, Matteo T. Degiacomi, Chris G. Willcocks*
Shape tracing: An extension of sphere tracing for 3D non-convex collision in protein docking
2020 IEEE 20th International Conference on Bioinformatics and Bioengineering (BIBE), 2020, DOI: `10.1109/bibe50027.2020.00016`

Chapter 5 discusses our molecular modelling work from a collaboration that aimed to provide better insight into possible SARS-CoV-2 spike glycoprotein – ACE2 complexes in light of the COVID-19 pandemic.
authors TBD
Dimeric ACE2 crosslinks SARS-COV2-Spike protein both promoting and inhibiting infection
In preparation

Common Terms & Abbreviations

Bound A protein monomeric unit that has been obtained from a known, resolved complex; *i.e.* the protein is already in the necessary conformation to bind with its partner.

CAPRI Critical Assessment of Prediction of Interactions. The community of scientists that develop and test protein-protein docking software.

CCC Cross correlation coefficient. A statistic that measures the similarity between two identically sized sets of data. A value approaching 1 means the sets are identical, while -1 indicates the opposite.

$f_{\mathbf{nat.}}$ Fraction of correct residue contacts. This is the ratio of correct residues in contact (defined as an atom within 5 Å of the partner), between a predicted pose and the target structure. A value of 1 indicates that we have exactly predicted the binding interface, while 0 means otherwise.

Ground Truth Also referred to as the known docked bound position or target structure. Meaning the quaternary structure of the protein complex, with the various monomeric units in their correct, respective positions. The goal of protein docking is to take two unbound proteins and predict the ground truth from those.

JabberDock Our *de novo* protein docking engine that harness STID isosurfaces to generate a predicted complex formed by two monomeric binding partners.

MD Molecular Dynamics. *in silico* modelling to generate trajectories of atoms by following Newton's equations of motion, using a forcefield to determine the interaction profiles.

PDB Protein Data Bank. Also refers to the unique 4 letter code given to an atomic structure available on the PDB webserver.

PES Potential Energy Surface. Describes the energy of a system associated with a set of parameters. In our protein docking work, the energy, or score, is a function of the rotations and translations applied to a ligand.

Pose Also referred to as docked position. Meaning a specific, structural arrangment of two binding partners.

PSO Particle Swarm Optimisation. An optimisation technique that seeds n particles onto a potential energy surface, before exploring the search space according to a particles current position and velocity. The movement of a particle is influenced both by its local position and of the best positions found by other particles.

RMSD Root-mean-square deviation. A measure of the average distance between two sets of atomic coordinates, for example between a target receptor-ligand complex and a predicted receptor-ligand complex.

RMSD_I The interfacial RMSD between two complexes, using only the α carbons at the interface (within 10 Å of the partner) of the ground truth.

RMSF Root-mean-square fluctuation. Similar to RMSD, it measures the distance fluctuation from some well-defined average position on a per-residue basis, *i.e.* in the context of this thesis, it is calculated from a singular trajectory of a structure.

SASA Solvent-accessible surface area. The surface area of some molecule accessible by some solvent, the size of which is determined by probe size – typically 1.4 Å to represent water.

SDF Signed distance function. The signed distance function of a set provides the distance from some arbitrary point to the boundary of the set. For our work, the SDF of a point gives the distance to the boundary of the STID isosurface. The function has positive values inside the isosurface, and negative outside.

STID Spatial and Temporal Influence Density. Referring to either a STID map: a volumetric grid of points with discrete values ranging from 0 to 1 (larger values roughly equating to greater local mass or charge), or a STID isosurface: a shape taken at an isovalue from the map. STID maps are our novel means to represent a protein in silico, capable of simultaneously accomodating side-chain dynamics, electrostatics and protein shape.

Unbound A protein that is known to participate in the formation of some complex with a binding partner, yet has been resolved independently from the partner. Unbound protein docking is the true test of a protein docking engine, as predicting the complex A + B → CD is far more challenging than A + B → AB.

1 Introduction

1.1 The building blocks of life

If atoms are the building blocks of the Universe, then amino acids are the building blocks of life. Underpinning the phenomena of life as we know it, from the single-cell organism to a human, is the ability of these amino acids to assemble into chains. These chains can be anywhere from a dozen amino acids to the thousands or tens of thousands. Typically, the smaller molecules ($\lesssim$ 50 amino acids), or those produced through post translational modifications or cleavage, are called peptides, although the exact definition of a peptide according to its size is ambiguous. The larger chains we call proteins, deriving from the Greek *proteios*, meaning "first" or "foremost", reflecting their importance to life. The sequence in which the 20 standard amino acids are arranged determines how the protein will fold into a 3D shape, and is therefore responsible for the secondary, and subsequently tertiary and so forth, structure (see Figure 1.1). It is the specific interactions permitted by the structure of these proteins that facilitate most of the work vital for the regulation of an organism. Thus, *the structure determines the function* of a protein.

When a protein misfolds, whether as a consequence of a mutation (*i.e.* the sequence fundamentally changes) or entropy over a host's lifetime, the result is usually benign, and the protein is destroyed. Occasionally, however, these misfolded proteins can propagate and grow in number, becoming the dominant variant. Proteins can also retain their original structure following a mutation but have their binding affinities to different partners altered. Major diseases such as Alzheimer's,[1] diabetes,[2] Parkinson's[1] and cancer[3] are as a consequence of a protein's operational malfunction. Proteins also play a large role in a range of infectious diseases, making them partly responsible for devastating pandemics such as COVID-19.[4] Resolving the molecular structure of a protein is, therefore, an area of critical research, both from a medical perspective, but also to enhance our understanding of the proteome and subsequent higher-level biological functions. The Protein Data Bank (PDB)[5] is a testament to this, with over 170000 structures having been deposited to date. The PDB was originally established in 1971 containing only 7 structures, but has since grown exponentially into the world's largest and most widely utilised databank of protein structures, with 14058 structures released in 2020 alone.[6]

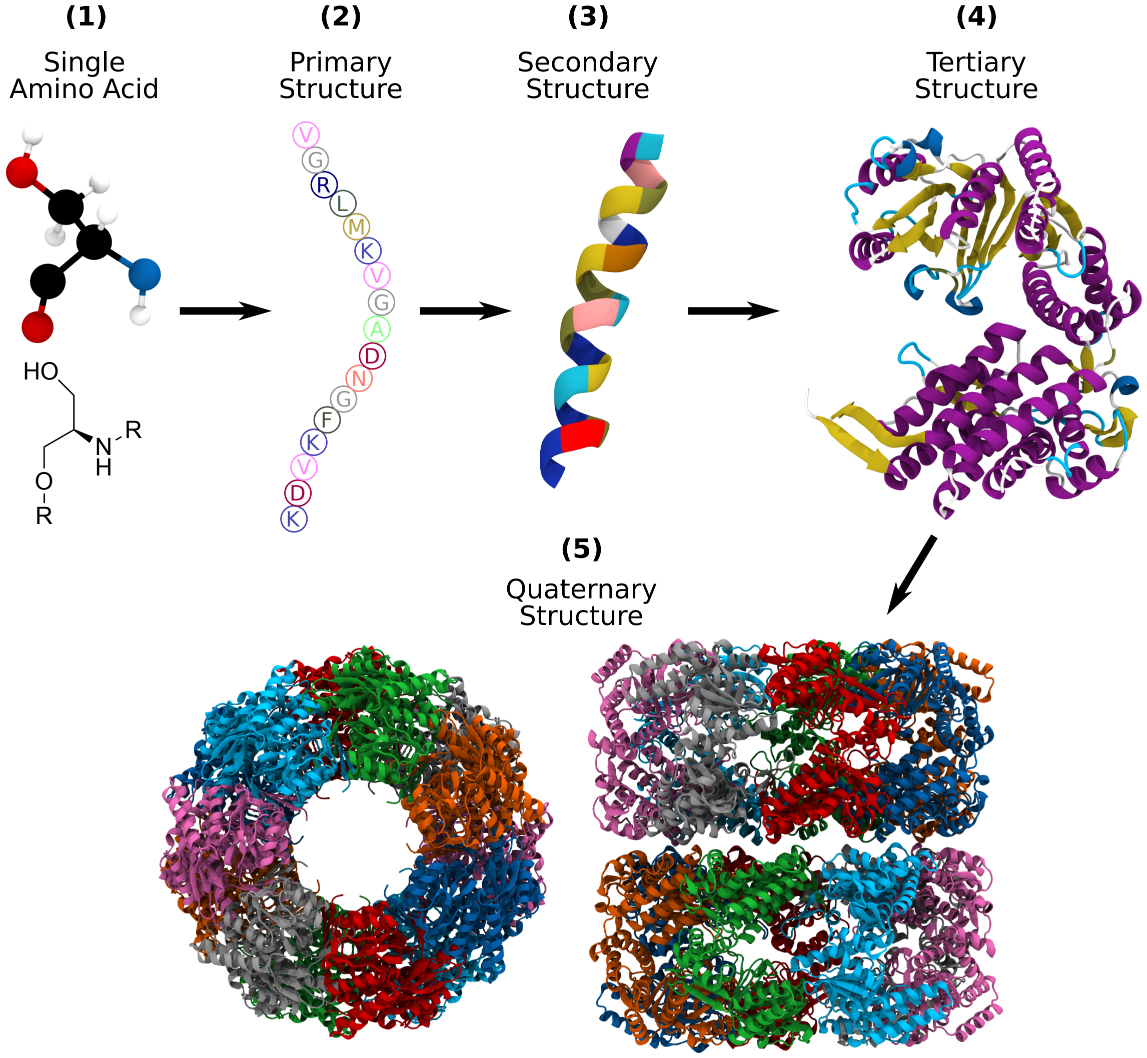

Figure 1.1: Protein structure scales for GroEL (PDB: 1SS8). **(1)** Individual atomic elements make up a single amino acid, in this case, serine. R denotes the continuation of the backbone. **(2)** Amino acids join into a chain to form the sequence/primary structure, coloured by amino acid type. **(3)** The chain folds into a 3D secondary structure, a local arrangement of amino acids (coloured by type) adjacent to one another in the sequence. Here, the chain folds into an α-helix. **(4)** Different regions of the larger chain fold heterogeneously, giving rise to a variety of secondary structural elements – β-sheets (yellow), helices (purple), turns (cyan), and unstructured (white). These, together, make up the tertiary structure of the protein. **(5)** Different chains can assemble into a quaternary structure. Here, we colour the complex by chain.

1.2 Resolving the structure of a protein

Resolving the structure of a protein is non-trivial, with several distinct techniques having evolved over the years to address the issue.[7] X-Ray crystallography is by far the most popular, providing ~89% of the protein structures[8] in the PDB[5] at the time of this thesis' submission. While the technique is currently the most widely used, ensuring the proper crystallisation conditions is complicated and time-consuming; indeed, the necessary crystallisation procedure can occasionally lead to conformational changes away from the native state,[9] resulting in misleading structures being deposited in the database. Furthermore, the method struggles with proteins that lack a well-defined structure, such as intrinsically disordered proteins, due to the requirement of a well-ordered crystal. Thus, the portion of the proteome largely available on the database are weighted towards less dynamic proteins that are easier to isolate and characterise, dynamic referring to anything from side-chain motion to considerable backbone and domain reshuffling. Nuclear Magnetic Resonance (NMR),[10] making up 7% of the database, can produce an ensemble of structures, providing key information on the dynamics of a protein and conformational flexibility. However, to uniquely determine a structure requires a variety of NMR experiments, and in general, these methods are unable to resolve larger complexes (typically ~50 kDa), as NMR is essentially an optimisation problem, with the number of interatomic distances, used ultimately to resolve the structure, increasing as $O(n)$ with n atoms. Finally, cryo-Electron Microscopy (cryo-EM),[11] making up 3% of the PDB, has increasingly gained traction with the community due to its significant benefits of providing native-like environments and not requiring challenging crystallisation conditions. However, routinely obtaining high-resolution structures comparative to NMR and X-ray crystallography has remained elusive due to current microscope resolutions. Furthermore, cryo-EM is typically restricted to large complexes ($\gtrsim$ 100 kDa). Despite these barriers, it is widely expected that cryo-EM will become the primary tool in protein structure determination due to its ease of use and consistent improvement in the quality of structures through the continuing "resolution revolution".[7]

In general, our concept of the proteome is skewed by the techniques used to observe it,[12] thus the majority of resolved structures are rigid molecules, placing limitations on our interpretation of a protein's conformational space and dynamics in its function. Another impact of this bias is our limited access, and therefore understanding of, membrane proteins, whose reliable structure determination continues to evade all three methods. Of the ~170000 structures available on the PDB, only ~7000 (4%) are transmembrane proteins[5] despite them making up 20-30% of the proteome and 50% of all known drug targets.[13] This relatively small number of available structures is primarily due to the greater technical difficulties associated with characterising them compared with soluble proteins, whereby removing them from the lipid-soluble environment can destabilise them. NMR[14] and Cryo-EM[15] have shown to be able to overcome some of the issues encountered by X-ray crystallography,[16] but both still have their limitations.

Integrative modelling (see Figure 1.2), where one or more techniques are combined to fill knowledge gaps or enhance our confidence in a structure, has increasingly been used to address the shortfalls of these methods, particularly for problematic protein complexes.[7,17] For example, low-resolution electron density maps generated from X-ray crystallography or cryo-EM can still prove useful when coupled with other structural data. Where NMR is unable to provide a full structure, it can provide sparse data on structural restraints; namely secondary structure and torsion angles from chemical shifts,[18] orientational restraints from residual dipolar coupling (RDC),[19] and distance restraints from Nuclear Overhauser effect (NOE) experiments.[20] Other experimental techniques have developed over the years to address the limitations of these three key methods, although none can generate a structure by themselves. Mass Spectrometry (MS) provides structural information through the mass of the protein or protein fragments. Clever MS pre-processing techniques can provide a rich amount of structural information; chemical cross-linking[21] yields information on residue contacts and protein conformational dynamics, Hydrogen/Deuterium Exchange probes the backbone structure, while covalent labelling[22] informs on side-chain flexibility and solvent accessibility. Additionally, ion mobility offers data on the shape of the protein or protein complex. A significant advantage of MS is that it does not suffer from the mass limit issues, and requires only a small sample size – in contrast to the techniques mentioned above. Electron paramagnetic resonance (EPR) spectroscopy, similar to NOE experiments in NMR, can provide distances between individually labelled atoms,[23] as well as identify secondary structure elements.[24,25] Small-angle X-ray scattering (SAXS) can identify the oligomeric state of a protein, and provide details on the shape of a complex while retaining its solution environment.[26] Generally, SAXS is used to provide low-resolution data; however, recent efforts have demonstrated that it can be used exclusively with the sequence to provide the structure.[27] Förster resonance energy transfer (FRET) can be used to probe the specific interactions of different subunits of a protein, including long-range interactions, by attaching an acceptor (usually a yellow fluorescent protein), and donor (cyan fluorescent protein) fluorophores at different termini. It has the distinct advantage in that it can be performed both *in vivo* and *in vitro*, providing key information on both individual molecule[28] and protein-protein interactions.[29] All of these methods, by themselves, are unable to elucidate the entire protein structure. Still, by integrating the limited structural elements from each, we can overcome their individual limitations, thereby resolving the structure of a protein.

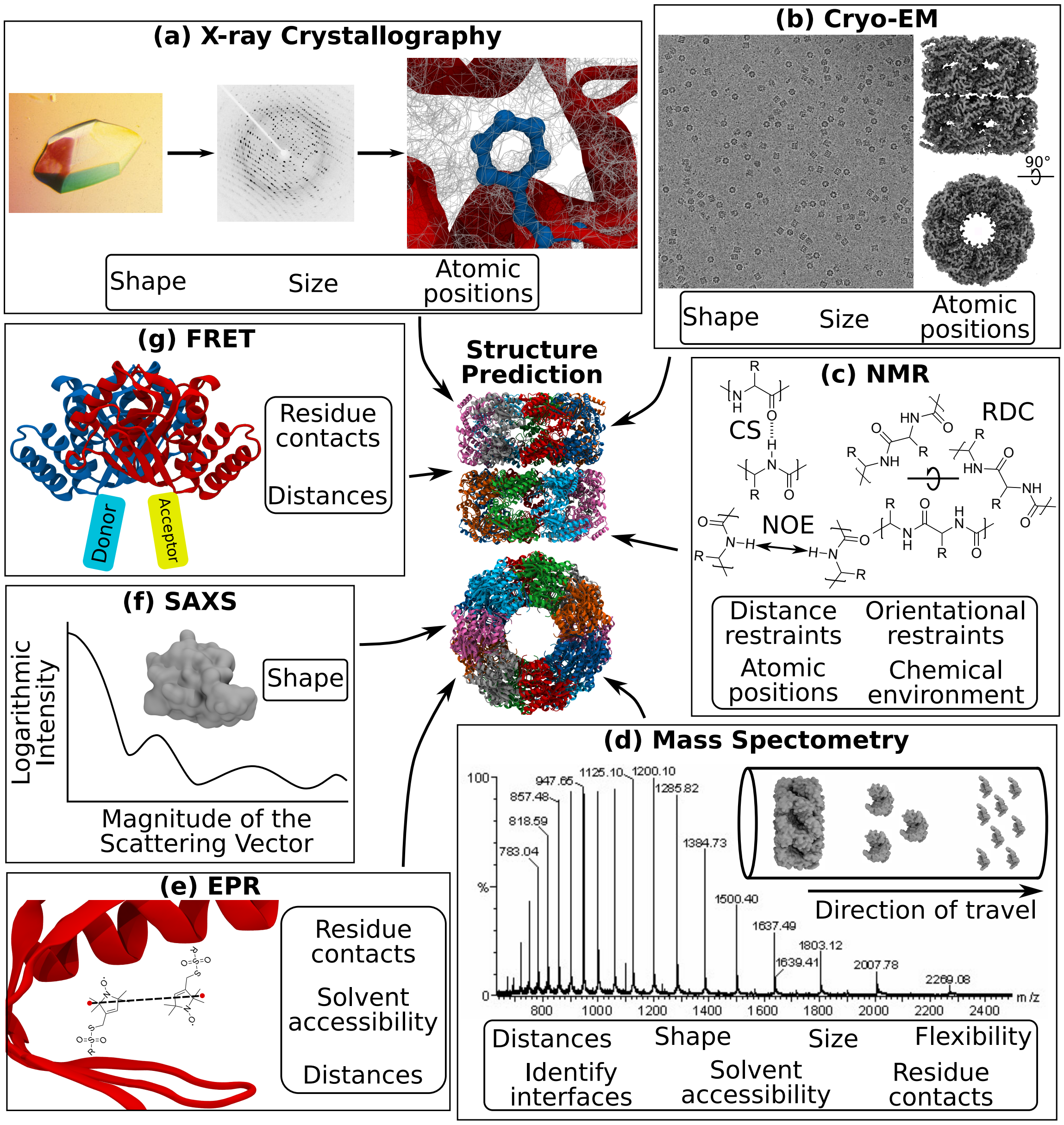

Figure 1.2: Schematic of different experimental techniques that can be integrated to enhance protein structure prediction. Each subfigure contains the specific structural information that can be extracted from the technique. All protein renderings, including domains and subdomains, are of GroEL (PDB: 1SS8, EMDB: 8750), except for the SAXS subfigure. **(a)** X-ray crystallography. An X-ray diffraction pattern[30] of a crystallised protein[31] is Fourier transformed into an electron density map. Atomic structures are refined against this map. **(b)** Cryo-Electron Microscopy. Thousands of proteins images are extracted from a micrograph[32] taken in a flash-frozen solution; the combination of these images provides an electron density map. **(c)** Nuclear Magnetic Resonance. Inter- and intramolecular interactions perturb the local chemical environment, providing structural data from CS, RDC, and NOE experiments. **(d)** Mass Spectrometry. Protein or peptide fragments have different time of flights according to their mass/charge ratio, providing distinct peaks in the spectra.[32] **(e)** Electron Paramagnetic Resonance. Paramagnetic spin labels, most commonly nitroxide, are attached to cysteines and the distances between their average position measured. **(f)** Small-angle X-ray scattering. The scattering profile of a protein (here lysozyme, PDB: 1DPX) in solution can be used to determine an average shape. **(g)** Förster Resonance Energy Transfer. Donor and acceptor fluorophores are attached to the termini of two proteins of interest to probe the protein-protein interactions.

All of these experimental techniques rely, at some level or another, on computational software to interpret the experimental data, and there are a plethora of existing means to assist users in decoding their raw results. Both X-ray crystallography and cryo-EM utilise human-guided spatial fitting algorithms to position atomic structures into electron density maps; popular software includes MAUD,[33] and Situs[34] respectively. Chemical shift NMR predictive software, such as PROSHIFT,[35] requires making chemical shift predictions from a model, before assessing and matching with experimental data. CHESHIRE,[18] can combine this data with the sequence to predict backbone torsion angles and subsequently secondary structure predictions from PDB insertions. The Integrative Modelling Platform (IMP)[36,37] offers a range of interpretive software to perform integrative modelling. Recent tools designed within the IMP includes a means to generate coarse-grained models of large protein complexes to match, score and rank against ion mobility data of protein shapes.[38,39] The popular Rosetta suite also offers a comprehensive selection of software for most techniques to facilitate integrative modelling. For instance, CS-Rosetta[40] provides chemical shift-based predictions, RosettaNMR[20] integrates NOE and rotational restraints into a structure prediction scoring function, while RosettaEPR[23] has been developed to extract data from EPR. The perpetual advancement of these integrative computational methods continues to enrich the exploitation of raw data in the pursuit of more accurate structure prediction.

These computational interpretive tools generate protein models to match and score against the experimental data. This procedure gives confidence in the model and allows the software to change and improve on it iteratively. However, many of these models struggle to consider the impact of protein dynamics on the experimental data. The aforementioned Situs software,[34] operates by projecting an atomic structure probe onto a lattice, convolving this with a Gaussian kernel, before attempting to match the generated density with a low-resolution electron density map from either cryo-EM or SAXS. They use Fast Fourier Transform (FFT)-accelerated methods to rapidly sample the conformational space, determining both the best fit and whether the probe is appropriate. This method, inherently, struggles to consider the impact of native dynamics, although recent strides to include flexibility have been made.[41] Typical approaches to characterising protein shapes through collision cross-section measurements *via* ion mobility, involve generating a super coarse-grained representation from a crystal structure and matching the collision cross-section of this convex shape against the experimental data. Since the experimental data includes information on dynamics, there is potential here for improved matching. Recent efforts, such as IMMS_Modeler[42] within IMP, have begun to better consider the impact of dynamics on generating predictions *via* a simulated annealing Monte Carlo procedure. Similar approaches have also successfully used Molecular Dynamics to give better consideration to the dynamics and flexibility.[43] Consideration of protein dynamics is, therefore, critical to fully interpreting the experimental data, and it is only very recently that current tools have begun to include dynamics into their modelling.

In addition to providing models to match experimental data, computational methods have also been used to generate key parameters used in experimental techniques. One such fundamental biophysical quantity is the density of a protein. Protein mass density; *i.e.* the density of an individual protein as opposed to the density of a protein solution, is an essential parameter for a range of experimental techniques, most notably X-ray crystallography structure determination. Since the late '70s, it has been assumed to be a constant with several proposed values derived from both experimental data and theoretical techniques.[44–48] In 2004 Fischer *et al.*[49] compared the work on the theoretical densities of 12 proteins calculated *via* Voronoi tessellation by Tsai *et al.*,[45] with experimental values from Squire & Himmel,[47] and Gekko & Noguchi.[48] They deduced that 1.410(6) g cm^{-3} should be taken as a minimum value and that the density is molecular weight-dependent. In contrast, protein densities derived from ion mobility measurements (*in vacuo*) have been found to be substantially lower, with values between 0.85 g cm^{-3} and 1.1 g cm^{-3}.[50] However, these gas-phase densities make geometric assumptions from the collision cross-section data, and it has been shown that these can lead to a reduction in the measured density.[51] In 2020 alone, values ranging from 1.23[52] to 1.5 g cm^{-3}[53] (with 1.35 g cm^{-3} being the most common) have been used in 22 different works, from those looking at collagen-water interactions[54] to protein film formation on cell culture surfaces.[55] The values proposed by Fischer *et al.* referenced in those works were primarily derived from computational models of static crystal structures. Since proteins are inherently dynamic, it is necessary to consider whether the consensus value for protein density holds in different environmental conditions, where we do not treat the protein atoms as fixed in space. Ashkarran *et al.*[56] recently presented a novel technique using magnetic levitation to measure the density of proteins in solution. They indicate that the density could be substantially lower (at a value of 1.03(2) g cm^{-3}) than previously thought. The authors hypothesise that the observed distribution and disagreement with the hitherto accepted value may be due to protein dynamics. Ensuring that the density of a protein used in a wide range of fields is correct is of clear benefit to the community, but this depends on a more reliable computational model that better considers the native dynamics of a protein in any environment.

1.3 *In silico* protein structure prediction

In addition to allowing structural biologists to interpret their data, *in silico* methods can independently complement experimental methods, or indeed provide insight not available through *in vivo* or *in vitro* experimentation. These computational tools broadly fall into three catagories: protein folding,[57] Molecular Dynamics[58] and protein-protein docking.[59]

1.3.1 Protein folding

Despite the significant strides made by the community, there is a considerable gap between the number of deposited protein structures in the PDB and the number of known sequences (185 million) in the UniProt database.[60] Predictive protein folding is the solution to this (Figure 1.3). It involves generating a protein's secondary and tertiary structure from its primary structure/sequence. The protein folding community is nucleated around the Critical Assessment of protein Structure Prediction (CASP)[61] biannual competition, which attempts to identify the most competitive means to predict protein structure. While CASP includes prediction categories for integrative modelling with experimental data from FRET, NMR, SAXS and cross-linking MS,[62] it also offers opportunities to demonstrate the viability of *ab initio* methods. These software, such as Rosetta,[63] I-TASSER[64] and QUARK,[65] are primarily based on Monte Carlo methods that sample possible backbone conformations using sections of the input sequence to insert structural fragments from the PDB iteratively. An associated scoring function based on various physical and empirical terms is then used to score a specific structural arrangement. The task is made considerably more manageable through homology modelling, whereby a similar sequence with a known structure is used as the basis for prediction. Popular homology modelling tools include Modeller[66] and SWISS-Model.[67]

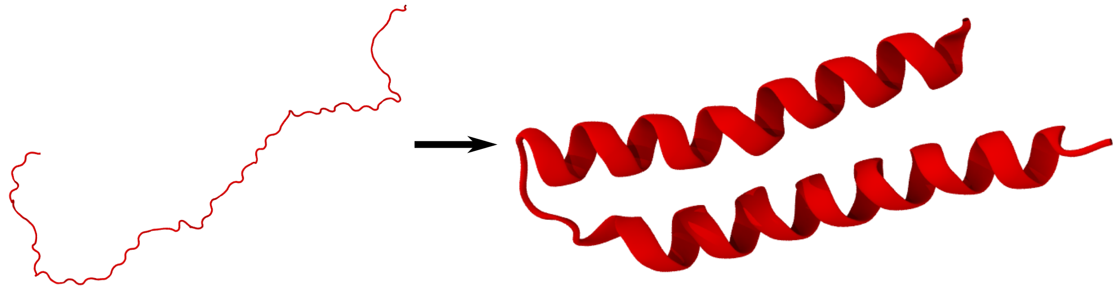

Figure 1.3: Protein folding example: from an unstructured species into two α-helices. The two α-helices on the right are extracted from a monomeric unit of halorhodopsin (PDB: 1E12), with the structure on the left an unstructured prediction of the same sequence returned by Modeller[66] without any structural constraints.

In a very recent watershed moment for the CASP community, Google's AI-based AlphaFold[68,69] achieved a median score of 92.4 Global Distance Test (GDT) for the CASP14 test set.[70] GDT is a score between 0 – 100 related to the percentage of amino acids within some small threshold distance from the correct position. A score of 90 GDT is considered competitive with experimental methods, a threshold other software rarely meet. This score is a vast improvement from traditional physics-based approaches, leading to the CASP assessors themselves to conclude a general solution to the 50 year protein structure prediction problem. It is important to note, however, that it is not yet a universal solution, as AlphaFold was unable to provide accurate models for the most flexible cases – although it did achieve a median score of 83.5 GDT for the hardest cases *versus* 67 GDT from the best human-assisted teams. Furthermore, AlphaFold relied on a training set made up of structures from the PDB, which, as we have discussed, is biased towards structures that are easier to crystallise. Thus, edge-case protein families and potentially less understood areas of the proteome, including transmembrane proteins, still need thorough benchmarking before a universal solution is declared. Nevertheless, this is still a remarkable scientific breakthrough.

1.3.2 Molecular Dynamics

Proteins often adopt multiple conformations *in vivo* to fulfil their function, and indeed will usually react to some binding interaction through an allosteric conformational switch. Many of these experimental techniques, particularly X-ray crystallography, offer only the static structure of protein in a singular conformation, neglecting its dynamics from the side-chain motion to larger domain-level shifts. Capturing these dynamics is, therefore, critical to understanding the function of a protein. Molecular Dynamics (MD) offers an *in silico* means to sample the dynamics of a protein at an atomistic resolution. MD is versatile, allowing users to create complex heterogeneous systems with many interacting components, for example, the SARS-CoV-2 spike glycoprotein with attached glycans in a representative viral membrane bilayer.[71]

Classical MD typically requires the use of a physics-based forcefield, which provides properties such as charge and mass to individual atoms (or beads in a coarse-grained representation), and other key parameters for bonded and non-bonded potentials. Forcefield terms are typically derived either from experimental data,[72] quantum calculations,[73] or recently machine learning methods.[74] Additional measures such as a thermostat or barostat are applied to ensure the system remains thermodynamically stable and accurate while a trajectory of new structures are generated *via* Newtonian mechanics. While various MD engines, such as GROMACS[75] and NAMD[76] exist, each offering different tools for complex modelling, the primary consideration for an MD simulation is the choice of forcefield. Numerous forcefields, such as Amber,[77] CHARMM,[78] and GROMOS[79] have evolved over the last 30 years to match specific thermodynamic conditions and reproduce certain molecular properties such as vibrational frequencies,[73] incrementally leading to more accurate models of protein behaviour in a native-like system.

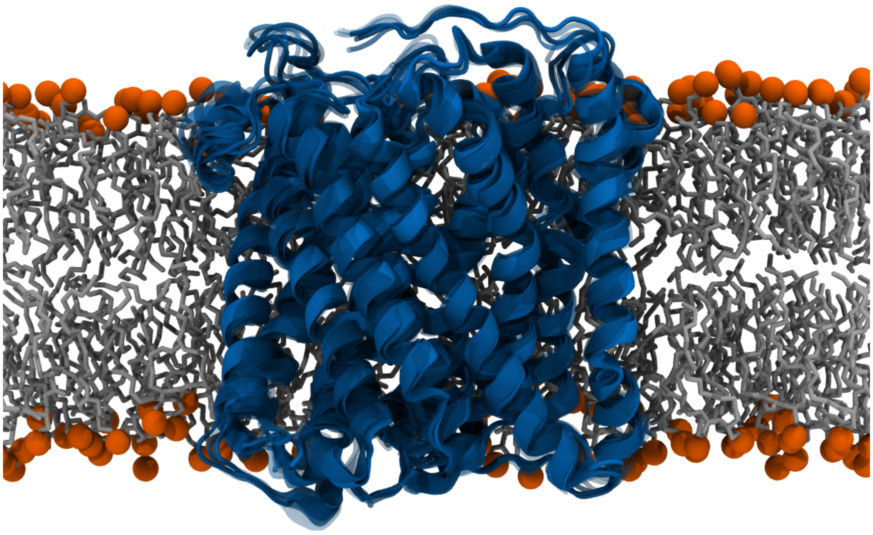

Figure 1.4: MD can provide high-resolution insight into the dynamics of a protein for various complex heterogeneous systems, such as this cytochrome c oxidase subunit (PDB: 3S33) in a lipid bilayer.

Progressive hardware advancements have allowed users to move from the picosecond timescale in the 1990s, to the microsecond in the present day. This computational power permits us to sample an extensive range of essential dynamics, from side-chain motion to previously inaccessible conformational states and kinematics; for example, the influence of substrate and inhibitor binding to the structure of the human CYP2D6 wild-type enzyme.[80] Consequently, MD has had a significant impact on medical studies. For instance, it is regularly used to identify binding sites and provide related binding energies for protein-drug interactions over biologically relevant timescales.[81] In this, MD provides a clear advantage over a structure from the PDB, which may not permit a specific ligand's binding due to its deposited conformation. In an integrative modelling context, MD has been combined with existing data, such as NMR,[82] to constrain a system to particular conditions over long timescales which would not usually be viable under free dynamics.

Despite the insight offered by MD, there are limitations to the method. Since the computational cost of an MD simulation is heavily dependent on the number of atoms and hardware being used, we are only now reaching the millisecond timescale for the smallest of proteins using specialised hardware,[83] hardware which is often only available to exclusive groups. In contrast, allosteric regulation in response to binding events regularly occurs over a multi-millisecond timescale,[84] while protein folding can take between hundreds of nanoseconds to seconds,[85] hence the need for predictive software. Lipid diffusion constants are typically on the scale of $\mathrm{nm}^2\ \mu\mathrm{s}^{-1}$,[86] adversely impacting the study of membrane protein systems. Coarse-grained methods provide an avenue to study these phenomena while avoiding the costs, but they are less widespread, and indeed less thermodynamically accurate than atomistic methods, with MARTINI[87] the only viable forcefield option for most. Even if a millisecond timescale is routinely achieved, the thermodynamic stability of a forcefield at this timescale is questionable.[88] Of additional concern is that MD is essentially non-ergodic in practice.[89] While a protein might sufficiently sample its phase space over microseconds *in vivo*, free MD may require second-long simulations to achieve ergodicity. This contrast is especially true for large proteins, and when there is significant conformational flexibility. Improved forcefields and hardware may brute force a solution to these problems eventually, but there are alternative solutions. Enhanced sampling MD techniques such as replica exchange offer a means to study long-timescale or rare events while calculating free energies.[90] In addition, recent efforts have highlighted the advantages of machine learning based methods.[74] More specifically, deep learning has shown to be able to provide plausible conformations distal from anything seen through free MD.[91] While these deep learning methods are still in their early stages, it is clear that they can enhance our understanding of a protein's phase space, and compensate for the issues encountered by MD. Still, the proteome's vast complexity necessitates that MD will always have a place in the study of conformational dynamics. Indeed, as well as being used to enrich studies of complex biological scenarios, MD is frequently used to consider dynamics in computational structure prediction; for example in conjunction with Modeller and cross-linking MS data to generate and filter models based on reasonable physics.[92] In this context, integrative modelling of MD into other *in silico* techniques is becoming increasingly powerful.

1.3.3 Protein-protein docking

Besides the tertiary structure and dynamics of a protein, predicting the quaternary structure forged by two interacting binding partners is the final key to rationalising a protein's function. To achieve their biological task, proteins often form homo- and heteromultimeric complexes. Given the plethora of genetic diseases associated with mutations that alter protein structure and consequently their capacity to interact with their binding partners,[93] a significant body of research and development focuses on methods for the elucidation of protein quaternary structures. In this context, computational techniques devised to predict the complex formed when two or more proteins bind can be of great help. Docking proteins *via* MD is possible in principle, but it is prohibitively expensive due to the number of degrees of freedom and the timescales over which docking events occur, although MD has been used to dock proteins with small drug molecules.[94] Protein docking engines are significantly cheaper and quicker than MD based methods, and their predictive capability can be harnessed to guide subsequent targeted experiments, both *in vitro* and in MD. Protein docking algorithms can leverage on integrative modelling to impose constraints, but rather more impressive are the docking engines capable of taking two binding partners and blindly predicting the correct complex from the tertiary structure alone. Protein-protein docking, in general, is the least understood or successful of the three *in silico* protein structure prediction methods, although its fulfilment would be another scientific breakthrough, facilitating a wide range of medical research such as drug discovery and therapeutics. Analogous to CASP, the community of protein-protein docking is centred around the Critical Assessment of Prediction of Interactions (CAPRI) competition,[95] which hosts two to four annual prediction rounds to determine the most competitive protein docking software.

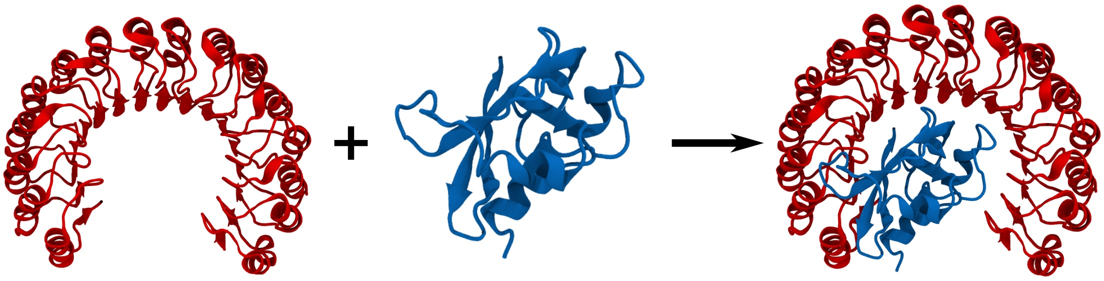

Figure 1.5: Predicting the assembly of ribonuclease inhibitor complexed with ribonuclease A (PDB: 1DFJ) from its two monomer units.

Protein-protein docking is a highly complex optimisation problem, requiring the generation of a considerable number of candidate arrangements. To accurately discriminate between correct and incorrect docked poses, a suitable scoring function is essential. Typical scoring functions used in this context involve a set of non-trivial physical or empirical terms, combined with custom weightings. An ideal scoring function should feature minimum mathematical uncertainty while accounting for protein structure and dynamics. A highly efficient exploration method is also required to navigate the landscape of possible conformations in search of the specific arrangements that minimise this scoring function. The two are intimately linked, with the scoring function guiding the behaviour of the navigator.

The simplest and most widespread approaches for protein-protein docking involve a global, systematic, rigid-body docking search. Typically, a large number of solutions are generated from a pair of static molecular structures,[96–98] and a scoring function is then used to identify the most favourable arrangement. Treating proteins as fully rigid objects while simultaneously using a scoring function that accounts for each atom's specific position leads to a modelling process that is excessively sensitive to the specific packing of atoms at the interface. To cater to protein dynamics, alterations to how models are built or how the scoring function assesses a protein arrangement are required. Possible strategies include rigid-body docking of ensembles of structures,[99–101] additional refinement stages that take place after rigid-body docking,[102,103] scoring functions that feature soft potentials to allow minor molecular overlaps,[104,105] pseudo-coarse-grained protein representations,[106] docking subunits connected by potentials,[107] matching protein surfaces represented as a collection of patches,[108] using normal modes to account for flexible conformational switches,[109] and relaxing the interface of docking poses using techniques such as MD, Monte Carlo (MC), or simulated annealing.[104,106,110] Some methods, such as HADDOCK,[111] IMP[112] and RosettaDock[113] feature scoring functions that utilise a combination of terms that describe physical interactions and penalise models that do not recapitulate available experimental data. Some docking engines such as HADDOCK, are able to apply constraints based on experimental data, such as FRET, during the exploration of the potential energy surface that scores the individual arrangements.[29] Ultimately, the hardest challenge for these different approaches is protein flexibility. In both the benchmark that the CAPRI community provides to newly established protein docking groups,[114] and in the complexes given to challenge competitors during the rounds, there is an assortment of proteins with varying levels of flexibility. Specifically, they are split into three classifications: easy, medium and difficult, according to their inherent flexibility (see Section 3.2.3.1 for more details). It is the harder, more flexible cases that all current approaches struggle with. This is primarily because flexibility is usually considered during a post-processing step, or because any coarse-grained representations used for docking are derived from a static structure. Even when an ensemble of poses is used, they are still a fixed set of atomic coordinates, and poor consideration is given to the specific side-chain mobility at the interface. Integrating MD, therefore, during the actual docking procedure seems an attractive prospect to removing these obstacles.

The vast majority of these docking algorithms focus on only a section of the proteome by considering the docking of water-soluble proteins. As we have already discussed, resolving the structure of a transmembrane protein is significantly more challenging. Indeed, the inherent complexity is highlighted by the distinct absence of a CAPRI benchmark for transmembrane proteins. Thus, a protein-protein docking tool capable of predicting transmembrane protein complexes would significantly benefit protein structure prediction. While docking transmembrane proteins is facilitated by limitations on the search space imposed by the lipid bilayer, membrane docking algorithms must consider the impact of the lipid bilayer on a protein's recognition of a partner in tandem with the solvent. In this context, there are only a small number of tools currently available. MPDock,[115] utilising existing Rosetta sampling and scoring methods in an integrative modelling context, found a successful high ranking pose in three out of five applied bound complexes. Hurwitz *et al.*'s program Memdock[116] uses a traditional rigid docking, refinement, re-ranking method, with energetic terms representing the membrane's hydrophobic environment included in the final stage. Comparing the performance of Memdock and GRAMM-X[117] on 11 unbound complexes, the authors showed that the first yielded a success rate of 36.4% and the latter of 9.1%. Viswanath *et al.* used the DOCK/PIPER[118] docking algorithm with an additional re-ranking step that considered the membrane transfer energy, achieving a success rate of 36.6% for 26 unbound complexes. Testing other software on the same dataset, the authors reported success rates of 30%, 46.6% and 56.6% for ZDOCK+ZRANK,[96,119] CLUSPRO[120] and GRAMM-X,[117] respectively. All of these approaches were only tested against cases featuring α-helical transmembrane proteins. Koukos *et al.*, using HADDOCK[121] without any specific membrane protein optimisation, achieved a blind docking success rate of 19.2% on their dimeric unbound dataset of 26 complexes that included β-barrel, monotopic and α-helix proteins. Of these 26 test cases, 11 featured a pair of integral proteins as binding partners, and only three of these were unbound-ligand-to-unbound-receptor docking. The latter achieved a success rate of 36.4%. HADDOCK has also very recently been combined as a refinement tool with the LightDock[122] docking algorithm and tested against 18 transmembrane-soluble protein complexes, achieving a success rate of 61.1%.[123] Currently, MPDock has only been presented as a proof of concept, not yet designed for widespread use. The DOCK/PIPER membrane energy re-ranking tool is available for download, but it must be applied to models obtained independently. Memdock is usable as a webserver though requires input structures to have their solvent-exposed regions manually removed. The limited number of docking engines, in a niche field still in its early stages, has resulted in little consideration being given to the protein dynamics, particularly as it pertains to the different environment imposed by a lipid bilayer. Therefore, there is a need for an effective and user-friendly tool that can consider the impact of biphasic dynamics on the docking of transmembrane proteins.

1.4 Underlying methods

The following covers the theory and main aspects surrounding the fundamental techniques used and discussed throughout this thesis.

1.4.1 Molecular Dynamics

1.4.1.1 The equations of motion

Molecular Dynamics (MD) is an *in silico* technique designed to simulate the behaviour and interactions of many-body systems over time. Classical MD uses Newton's equations of motion to describe the trajectory an atom or molecule might take:

$$\overrightarrow{F}_i = m_i \frac{d^2 \overrightarrow{r}_i}{dt^2} = m_i \overrightarrow{a}_i, \tag{1.4.1}$$

where $\overrightarrow{F}_i$ is the force acting on particle i, m_i the mass of the particle, $\overrightarrow{r}_i$ its position, $\overrightarrow{a}_i$ is its acceleration, $\overrightarrow{r}_i$ its position, and t is time. $\overrightarrow{F}_i$ is found through the total potential energy, $U_{\text{tot.}}$, which in the simplest microcanonical case, is given as:

$$-\nabla_i U_{\text{tot.}} = \overrightarrow{F}_i. \tag{1.4.2}$$

$U_{\text{tot.}}$ is found by summing the interactions on particle i from all other particles in the system. These interactions are determined through a forcefield, which are discussed in more detail in Section 1.4.1.2.

Several algorithms exist to integrate these equations of motion, thereby providing the trajectory of the system. The general approach is to divide the desired simulation time into a series of timesteps of length Δt. The timestep size represents a tradeoff between efficiency and the build-up of numerical error, which manifests as growing inaccuracy in the molecules' behaviour as the simulation progresses. The typical choice of $\Delta t = 1$ fs in classical MD strikes a balance between these two and is roughly ten times shorter than the period of the highest frequency motion of an atomistic simulation, the bond vibrations. Therefore, all motions within an MD simulation can be captured with this timestep without excessive computation time. The simplest algorithm used for calculating the motion of the particles is the Verlet algorithm:

$$\overrightarrow{r}_i^{n+1} = 2\overrightarrow{r}_i^{n} - \overrightarrow{r}_i^{n-1} + \Delta t^2 \overrightarrow{a}_i^{n}, \tag{1.4.3}$$

where n represents the current timestep. The Verlet algorithm provides good energy and momentum conserveration and is time-reversable, however the presence of the Δt^2 term can cause a rapid build up of system instability due to the aforementioned numerical error. The more commonly employed algorithm, and the one used throughout this thesis, is the velocity-Verlet algorithm, which omits this term. For this, each integration cycle first consists of calculating the velocities, $\overrightarrow{v_i}$, at a midstep, $n + \frac{1}{2}$:

$$\overrightarrow{v}_i^{n+\frac{1}{2}} = \overrightarrow{v}_i^{n} - \frac{1}{2}\overrightarrow{a}_i^{n}\Delta t. \tag{1.4.4}$$

From this, we can calculate the positions at the next step:

$$\overrightarrow{r}_i^{n+1} = r_i^{n} + \overrightarrow{v}_i^{n+\frac{1}{2}}\Delta t. \tag{1.4.5}$$

After calculating the accelerations at the next step from the potential at these updated positions, we can then find the velocities for the next timestep:

$$\overrightarrow{v}_i^{n+1} = \overrightarrow{v}_i^{n+\frac{1}{2}} + \frac{1}{2}\overrightarrow{a}_i^{n+1}\Delta t. \tag{1.4.6}$$

In modelling proteins, we usually require them to be embedded in some solvent; in this thesis, either water or lipid. Enough solvent must be used such that the surrounding solvent can behave as a bulk. Explicitly modelling the solvent to the necessary dimensions required would demand a large number of atoms and thus is prohibitively expensive. Furthermore, it requires the non-trivial consideration of the system's interface with the surrounding "box". Periodic boundary conditions, wherein a small unit cell at the edges of the system are repeated *ad infinitum* in three dimensions, is a resolution to this. Periodic boundary conditions effectively allow us to model an infinite number of atoms, thereby modelling the bulk at a significantly reduced computational cost. Care is taken when using periodic boundary conditions to ensure enough solvent to avoid any atoms interacting with their periodic image.

For a simulation to be physically sensible, it must be undertaken under reasonable, controlled thermodynamic conditions. These conditions are known as ensembles, and in this thesis, we will exclusively use two ensembles: (1) the Canonical, with a constant number of atoms (N), volume of system (V), and temperature (T); and (2) the isothermal-isobaric, with constant N, system pressure (P), and constant T. In the NVT ensemble, the pressure can vary to maintain the volume, while in the NPT ensemble, the opposite is true. We typically use the NVT ensemble in the simulation's equilibration phase and the NPT in the production, following a short equilibration in NPT. These ensembles rely on a thermostat and barostat (in the NPT case) to maintain their thermodynamic conditions. This can be done by, for example, rescaling velocities with the Berendsen thermostat, or by applying drag in the equations of motion using the Langevin thermostat. The Berendsen approach is typically used for equilibration, as it is both cheap and able to rapidly achieve system equilibration, but due to the rescaling is not deterministic. In contrast, Langevin, or other common methods like Nosé-Hoover, are deterministic but tend to be more expensive and therefore reserved for MD's production cycle.

1.4.1.2 Forcefields

In classical MD, we approximate the total energy of the system by splitting it into a series of different contributing energetic terms:

$$U_{\text{tot.}} = U_{\text{bond}} + U_{\text{angle}} + U_{\text{dihedral}} + U_{\text{elec.}} + U_{\text{vdW.}}. \tag{1.4.7}$$

U_{bond}, U_{angle}, and U_{dihedral}, describe the bonded terms, while $U_{\text{elec.}}$ and $U_{\text{vdW.}}$ are the non-bonded terms. We will briefly discuss below the form these terms take within the AMBER[124] forcefield, which is used for all MD simulations throughout this thesis. It is the specific form of these energetic terms, and the parameters used within them that differentiate forcefields. Accordingly, forcefields will have their parameters derived through different means. For example, in AMBER,[124] the parameters were designed to match protein NMR data, while CHARMM,[78] also designed with biosystems in mind, had its parameters determined through quantum chemical calculations of interactions with water. The inclusion of additional terms that deviate from Equation 1.4.7 are typically reserved for forcefields that wish to mimic particular systems, and generally have poor transferability.

The bonded terms ensure that the overall correct molecular geometry is maintained, which is critical for systems like folded proteins. The bonded stretching term, U_{bond}, directly connects two atoms, i and j, and is described through a harmonic potential:

$$U_{\text{bond}}(l_{ij}) = K^{\text{b}}_{ij}(l_{ij} - l^0_{ij})^2, \tag{1.4.8}$$

where l_{ij} is the distance between i and j, l^0_{ij} their equilibrium length, and K^{b}_{ij} the bonded force constant. Note the presence of the i and j indices for each term. In other words, both the bonded force constant and the equilibrium bond length are unique between these two atoms, while also being unique to the forcefield. For example, AMBER ff14sb[125] uses an $l^0_{ij} = 1.4$ Å and $K^{\text{b}}_{ij} = 392459.2$ kJ mol^{-1} nm^{-1} between two α-carbons, while CHARMM36[126] uses an $l^0_{ij} = 1.5$ Å and $K^{\text{b}}_{ij} = 186188.0$ kJ mol^{-1} nm^{-1}. These differences are mirrored with all analogous parameters in the other energetic terms.

U_{angle} is the angle bending term and exists between two atoms, i and k, that are connected *via* two bonds through a third atom, j. It is also given as a harmonic potential:

$$U_{\text{angle}}(\theta_{ijk}) = K^{\theta}_{ijk}(\theta_{ijk} - \theta^{0}_{ijk})^2, \tag{1.4.9}$$

where θ_{ijk} is the angle between i, j and k, θ^{0}_{ijk} is their equilibrium angle, and K^{θ}_{ijk} the angle force constant.

U_{dihedral} is the sum of the proper and improper dihedral angle terms, and describes the energy between a group of four atoms. The proper dihedral term, U_{proper}, is associated with the dihedral energy between a series of connected atoms i-j-k-l, where the bonds are connected as shown. In AMBER,[125] it is given as:

$$U_{\text{proper}}(\phi_{ijkl}) = \frac{V^{m}_{ijkl}}{2}\left(1 + \cos(m\phi_{ijkl} - \phi^{0}_{ijkl})\right), \tag{1.4.10}$$

V^{m}_{ijkl} is the proper dihedral force constant for multiplicity m, and ϕ^{0}_{ijkl} the phase angle for torsional angle parameters. The improper dihedral, U_{improper}, provides the energy when the atoms are not connected in a simple series, such as when j is covalently linked to i, k and l. For example, in a planar molecule, the dihedral arises from the angle between the j-l line and the ijk plane.

$$U_{\text{improper}}(\psi_{ijkl}) = K^{\text{im.}}_{ijkl}\left(1 + \cos(m\psi_{ijkl} - \psi^{0}_{ijkl})\right), \tag{1.4.11}$$

where $K^{\text{im.}}_{ijkl}$ is the improper dihedral force constant, and ψ^{0}_{ijkl} the phase angle, taking the value of 180° for $m = 2$ (planer molecule), and 120° for $m = 3$ (tetrahedral molecule).

While retaining the correct molecular geometry for a simulation is essential, explicitly modelling bond vibrations restricts the possible timestep to 1 fs. Applying a constraint ensures the bonds lengths are correct while allowing larger timesteps, usually 2 - 5 fs, thereby reducing the computational costs associated with a simulation. The constraint applied throughout this thesis is the LINCS algorithm,[127] which operates by first projecting an updated bond onto the same bond from the previous step before correcting the length of the bond to the specified length. These constraints become more expensive to apply the more bonded connections there are in a system.

The remaining interactions between any two atoms in a system, both inter- and intramolecularly, are described through the non-bonded terms. These are split between the van der Waals potential, U_{vdw}, and the electrostatic potential, $U_{\text{elec.}}$. U_{vdw} effectively combines dispersion, steric repulsion and induction interactions and is given through a Lennard-Jones potential:

$$U_{\mathrm{vdW}}(r_{ij}) = \epsilon_{ij} \left[\left(\frac{r_{ij}^0}{r_{ij}} \right)^{12} - 2 \left(\frac{r_{ij}^0}{r_{ij}} \right)^{6} \right], \tag{1.4.12}$$

where r_{ij} is the distance between atoms i and j, ϵ_{ij} is the well-depth of the potential, and r_{ij}^0 is the position of the minimum. This potential therefore features a repulsive wall at $r_{ij} < r_{ij}^0$, which prevents atoms from overlapping, and an attractive well when $r_{ij} > r_{ij}^0$. At large distances, U_{vdW} tends to zero, so the Lennard-Jones potential is well suited to describe the interactions between atoms.

Finally, the electrostatic interaction is given through a Coulombic potential:

$$U_{\mathrm{elec.}}(r_{ij}) = \frac{q_i q_j}{4\pi\epsilon_0 r_{ij}}, \tag{1.4.13}$$

where q_i and q_j are the charges on atoms i and j respectively, and ϵ_0 the permittivity of free space (8.8542×10^{-12} m^{-3} kg^{-1} s^4 A^2). Both U_{vdW} and $U_{\mathrm{elec.}}$ typically have cutoffs applied to them, usually 10 - 12 Å, to avoid excess computation of negligible interactions. However, since $U_{\mathrm{elec.}}$ tends to zero as r_{ij} increases at a much slower rate than U_{vdW}, MD simulations employ additional methods to include long-range electrostatic interactions within explicitly using Equation 1.4.13. The most common algorithm is the particle-mesh Ewald (PME) approach, which splits short- and long-range electrostatic interactions, with short-range interactions calculated *via* Equation 1.4.13, and long-range interactions beyond the cutoff found through a summation in Fourier space, with the system split into a series of unit cells, that continue periodically through the periodic boundary conditions. This long-range summation is between the charge-density field of the central unit cell, and the average charge-density field of the surrounding lattice. PME's inclusion of long-range electrostatics greatly improves the accuracy of molecular simulations.

There also exist more complex forcefields that do not consider atoms as simple point charge models and instead include an extra particle attached by a spring to an atom, thereby introducing polarisability.[128] Indeed, there are also forcefields that are capable of modelling the formation and dissociation of bonds through complex interaction terms.[129] Given that the method by which our protein representations are built, which aside from the MD simulation itself, uses the partial charges provided by the forcefield, we elected to use the "simpler" classical forcefields. Both AMBER and CHARMM have recently been shown to be highly accurate in modelling protein behaviour;[130] however, given AMBER ff14sb is specifically catered to model protein side-chains accurately, it seemed the more suitable choice as side-chain behaviour plays a vital role in the shape of our representations.

1.4.2 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a stochastic optimisation technique that uses a population of candidate solutions to explore a search space, sampling a fitness function as they move. Each particle is initialised with a random velocity and position, and their motions are influenced by their current momentum, the fit at their current position, the previous position with the best solution, and the position of the current best found global solution. It was developed by Eberhart and Kennedy in 1995,[131] and is inspired by the motion of bird flocks and schools of fish.

Let each particle, i, have position P_i^n at iteration n, in an m-dimensional space:

$$P_i^n = [x_{0,i}^n, x_{1,i}^n, x_{2,i}^n, ..., x_{m,i}^n], \tag{1.4.14}$$

where x represents a coordinate in the m-dimensional space. Each position in this space will have an associated fitness value, which collectively define the Potential Energy Surface (PES) that is being explored. At initialisation ($n = 0$), both the position of particle i, and the velocity, V_i^n, are randomised. The velocity at iteration n, is accordingly given by:

$$V_i^n = [v_{0,i}^n, v_{1,i}^n, v_{2,i}^n, ..., v_{m,i}^n], \tag{1.4.15}$$

where v represents the velocity associated with each coordinate. Over progressive iterations, each particle's updated velocity is influenced by the position of the previous best solution, $P_{i,\text{best}}^n$, and the current global best solution of the particle swarm, $P_{globalbest}^n$. Each particle's subsequent position is therefore given by the current position and this influenced velocity:

$$P_i^{n+1} = P_i^n + V_i^{n+1}, \tag{1.4.16}$$

where V_i^{n+1} is given by:

$$V_i^{n+1} = wV_i^n + c_1t_1\left(P_{i,\text{best}}^n - P_i^n\right) + c_2t_2\left(P_{globalbest}^n - P_i^n\right). \tag{1.4.17}$$

The first term represents the particle's current inertia, the second is the influence from the particle's previous best position, and the last is from the current global best position. w, known as the inertia weight, determines the balance between exploration and exploitation, with smaller values facilitating better exploitation of known solutions and higher values leading to exploration outside of these solutions. $t1$ and $t2$ are unique random weightings for each particle and each iteration and introduce a stochastic element to the latter two terms. Finally, the c_1 and c_2 hyperparameters define the amount by which the particles are influenced by their own history *versus* the group's. The weightings are often modified throughout the optimisation, with the early stages giving particles higher inertia to facilitate exploration, while the latter stages focus more on exploitation.

As a consequence of this approach, PSO is able to sample a PES and converge to global solutions rapidly. As it does not use a gradient, unlike traditional optimisation methods, it does not require the problem to be differentiable. It is also easily parallelisable due to the swarm nature of the method, and is therefore very suitable for rapid optimisation. PSO is used for all protein docking performed in this thesis, with the exception of the majority of the work in Section 4.4, and is currently used within the current iteration of JabberDock available on GitHub.

1.4.3 Solvent Accessible Surface Area

The solvent-accessible surface area (SASA) is an essential biophysical quantity that represents the surface area of a biomolecule that is accessible by some solvent molecule. In this thesis, we employ the commonly utilised Shrake-Rupley algorithm,[132] also known as the "rolling ball" algorithm. There are other SASA methods available,[133,134] which tend to be quicker but are less accurate.

Given an appropriate probe size, typically 1.4 Å to represent water, the Shrake-Rupley algorithm draws a mesh grid of points at the probe size's length beyond each atom's van der Waals radius in a molecule. Each point is then checked to ensure it is not buried within neighbouring atoms, and if it is not, it is classified as accessible (see Figure 1.6). The number of accessible points is then multiplied by the surface area each point represents to give a total value for the accessible surface area of a solvent with the associated probe size.

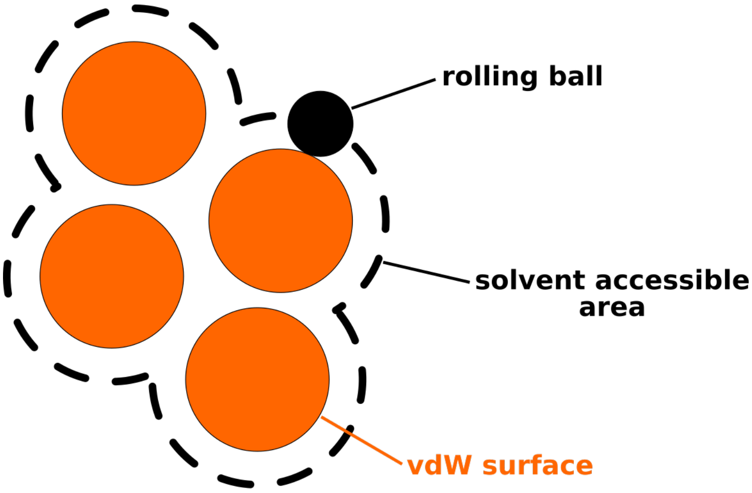

Figure 1.6: Schematic for how the SASA of a molecule is calculated. The black probe is "rolled" along the atoms' surface, itself represented as the van der Waals radius of each atom, tracing the path of the solvent accessible area from the position of the centre of the probe. Traced points within the molecule that clash with another atom are ignored, leading to only the surface accessible points considered in the SASA calculation.

1.4.4 Voronoi tessellation

Voronoi tessellation is a method to generate a series of n convex polygon cells in space that surround n points, where the dividing lines between cells lie equidistant between neighbouring points. An example of a Voronoi tessellated grid in 2D is shown in Figure 1.7.

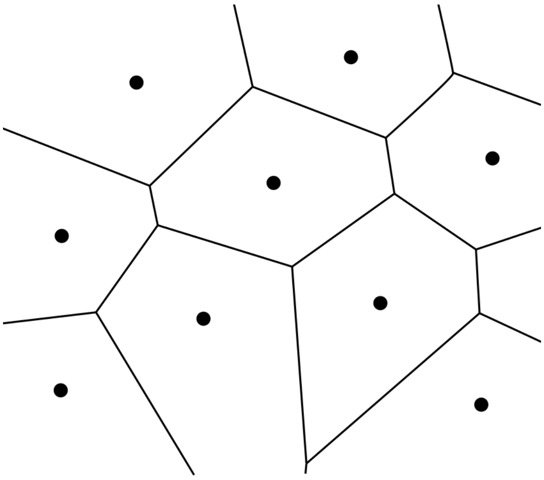

Figure 1.7: An example of Voronoi tesselation. Each point is placed in a cell, with the dividing lines of cells equidistant between points.

While Voronoi tessellation was not used explicitly in this thesis, it is discussed in the context of protein densities in Section 2.5, wherein it was the primary previous computational method to determine the density of a protein.[44–46] Here, each atom is treated as a point, and the cells are built in 3D around them. The density is subsequently calculated from the volume enclosed by this grid. As the atomic points must by necessity be fixed in space, Voronoi tesselation ignores the contribution of protein dynamics to density.

1.5 Thesis objectives & outline

Treating a protein as a rigid object is not sufficient to describe its dynamic properties, adversely impacting the computational interpretation of experimental data, and introducing a host of issues in a protein docking context. In this thesis, our primary focus will be on our solution to this issue; the Spatial and Temporal Influence Density (STID) map. The STID map is a novel pseudo-coarse-grained protein representation built from a Molecular Dynamics simulation, capable of simultaneously accomodating side-chain dynamics, electrostatics and protein shape. In general, however, this study's emphasis is to highlight the importance of accounting for dynamics in protein assembly, paving the way to enhanced protein structure prediction. The main aims and objectives of this thesis, therefore, are as follows:

1. To introduce a protein representation that encompasses the critical characteristics of a protein; its side-chain dynamics, electrostatics and shape, and then apply this representation to a series of physical and biophysical problems to prove its utility.
2. To apply this representation to the protein-protein docking problem, both for water and lipid-soluble species, further highlighting the importance of adequate side-chain motion accommodation.
3. To demonstrate how harnessing the conformational dynamics of microsecond long atomistic simulations can elucidate diverse quaternary structure, and indeed inform us on what is required from a larger *in vivo* system.

To this end, each Chapter will aim to accomplish the following:

Chapter 2: A novel representation of molecular structure and dynamics

This Chapter introduces the concept of the STID map. It lays out the physical background and provides a series of benchmarks to prove its utility. It describes how we can infer permittivity trends, calculate accurate protein densities in different environments, and extract information about the disruption of a residue's conformational dynamics from some distal binding interaction.

Chapter 3: JabberDock: Accounting for side-chain dynamics in protein docking

The aims of this Chapter are to utilize the STID map in a protein-protein docking context; embodied in our protein docking engine, JabberDock. We challenge JabberDock with the CAPRI benchmark's 224 soluble protein complexes, and 20 transmembrane complexes collated from a series of previous transmembrane protein docking attempts.

In this, JabberDock achieves a success rate of 54% and 75%, respectively. We close the Chapter by successfully applying JabberDock to two unsolved structure prediction problems, one water-soluble and one lipid-soluble.

Chapter 4: Beyond JabberDock

This Chapter explores possible steps JabberDock users can take to improve the quality of predictions, as well as refinements to the STID map and improvements to the optimisation routine used for docking. While not complete, we endeavour to establish the practicality of techniques borrowed from the Computer Graphics field in a protein docking context.

Chapter 5: Dynamics of the SARS-CoV-2 spike glycoprotein

In this Chapter, we move beyond the limits of side-chain dynamics and demonstrates how large, long-timescale simulations of proteins can assist in protein quaternary assembly prediction. Specifically, we take the example of the various oligomeric states formed between the SARS-CoV-2 spike glycoprotein and ACE2 receptor in the wake of the COVID-19 pandemic. From this, information about the necessary local curvatures required of both the virus and ACE2 host cell can also be extracted.

Chapter 6: Conclusions

Finally, this Chapter draws together the key results of the previous Chapters, concluding that better consideration of protein dynamics leads to better protein structure prediction and depth of understanding of a protein's function. We close with a brief look into our vision for future work.

Much of the material presented here has either been published in peer-reviewed journals, is currently under review or is under preparation for publication. Rather than present this material in its article format, it has instead been adapted for this thesis. For Chapter 5, this includes a large quantity of data not included in the final manuscript. Full details on which Sections have been integrated from publications or manuscripts are available in the Publications & Manuscripts Section following the Contents.

1.6 Bibliography

[1] M. T. Lin and M. F. Beal, *Mitochondrial dysfunction and oxidative stress in neurodegenerative diseases*, *Nature*, 2006, **443**, 787–795.

[2] N. Møller and K. S. Nair, *Diabetes and protein metabolism*, *Diabetes*, 2008, **57**, 3–4.

[3] N. J. Kelly, J. F. Varga, E. J. Specker, C. M. Romeo, B. L. Coomber and J. Uniacke, *Hypoxia activates cadherin-22 synthesis via eIF4E2 to drive cancer cell migration, invasion and adhesion*, *Oncogene*, 2018, **37**, 651–662.

[4] *WHO Coronavirus Disease (COVID-19) Dashboard — WHO Coronavirus Disease (COVID-19) Dashboard*, `https://covid19.who.int/`.

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

[6] *wwPDB: Deposition Statistics*, `https://www.wwpdb.org/stats/deposition`.

[7] J. T. Seffernick and S. Lindert, *Hybrid methods for combined experimental and computational determination of protein structure*, *J. Chem. Phys.*, 2020, **153**, 240901.

[8] A. Ilari and C. Savino, *Protein structure determination by X-ray crystallography*, *Methods Mol. Biol.*, 2008, **452**, 63–87.

[9] W. S. Ryu, *Molecular Virology of Human Pathogenic Viruses*, Elsevier Inc., 2016, pp. 1–423.

[10] J. M. Würz, S. Kazemi, E. Schmidt, A. Bagaria and P. Güntert, *NMR-based automated protein structure determination*, *Arch. Biochem. Biophys.*, 2017, **628**, 24–32.

[11] E. Nwanochie and V. N. Uversky, *Structure Determination by Single-Particle Cryo-Electron Microscopy: Only the Sky (and Intrinsic Disorder) is the Limit*, *Int. J. Mol. Sci.*, 2019, **20**, 4186.

[12] J. A. Marsh and S. A. Teichmann, *Structure, Dynamics, Assembly, and Evolution of Protein Complexes*, *Annu. Rev. Biochem.*, 2015, **84**, 551–575.

[13] J. G. Almeida, A. J. Preto, P. I. Koukos, A. M. Bonvin and I. S. Moreira, *Membrane proteins structures: A review on computational modeling tools*, *Biochim. Biophys. Acta - Biomembr.*, 2017, **1859**, 2021–2039.

[14] S. J. Opella and F. M. Marassi, *Applications of NMR to membrane proteins*, *Arch. Biochem. Biophys.*, 2017, **628**, 92–101.

[15] N. Thonghin, V. Kargas, J. Clews and R. C. Ford, *Cryo-electron microscopy of membrane proteins*, *Methods*, 2018, **147**, 176–186.

[16] I. Moraes, G. Evans, J. Sanchez-Weatherby, S. Newstead and P. D. Stewart, *Membrane protein structure determination - The next generation*, *Biochim. Biophys. Acta - Biomembr.*, 2014, **1838**, 78–87.

[17] A. P. Joseph, G. Polles, F. Alber and M. Topf, *Integrative modelling of cellular assemblies*, *Curr. Opin. Struct. Biol.*, 2017, **46**, 102–109.

[18] A. Cavalli, X. Salvatella, C. M. Dobson and M. Vendruscolo, *Protein structure determination from NMR chemical shifts*, *Proc. Natl. Acad. Sci. U. S. A.*, 2007, **104**, 9615–9620.

[19] C. A. Rohl and D. Baker, *De novo determination of protein backbone structure from residual dipolar couplings using Rosetta*, *J. Am. Chem. Soc.*, 2002, **124**, 2723–2729.

[20] P. M. Bowers, C. E. Strauss and D. Baker, *De novo protein structure determination using sparse NMR data*, *J. Biomol. NMR*, 2000, **18**, 311–318.

[21] T. Hofmann, A. W. Fischer, J. Meiler and S. Kalkhof, *Protein structure prediction guided by crosslinking restraints - A systematic evaluation of the impact of the crosslinking spacer length*, *Methods*, 2015, **89**, 79–90.

[22] V. L. Mendoza and R. W. Vachet, *Probing protein structure by amino acid-specific covalent labeling and mass spectrometry*, *Mass Spectrom. Rev.*, 2009, **28**, 785–815.

[23] N. Alexander, A. Al-Mestarihi, M. Bortolus, H. Mchaourab and J. Meiler, *De Novo High-Resolution Protein Structure Determination from Sparse Spin-Labeling EPR Data*, *Structure*, 2008, **16**, 181–195.

[24] W. L. Hubbell, H. S. Mchaourab, C. Altenbach and M. A. Lietzow, *Watching proteins move using site-directed spin labeling*, *Structure*, 1996, **4**, 779–783.

[25] M. A. Lietzow and W. L. Hubbell, *Motion of Spin Label Side Chains in Cellular Retinol-Binding Protein: Correlation with Structure and Nearest-Neighbor Interactions in An Antiparallel β-Sheet*, *Biochemistry*, 2004, **43**, 3137–3151.

[26] J. Köfinger and G. Hummer, *Atomic-resolution structural information from scattering experiments on macromolecules in solution*, *Phys. Rev. E - Stat. Nonlinear, Soft Matter Phys.*, 2013, **87**, 052712.

[27] C. Prior, O. R. Davies, D. Bruce and E. Pohl, *Obtaining Tertiary Protein Structures by the ab Initio Interpretation of Small Angle X-ray Scattering Data*, *J. Chem. Theory Comput.*, 2020, **16**, 1985–2001.

[28] M. Bonomi, R. Pellarin, S. J. Kim, D. Russel, B. A. Sundin, M. Riffle, D. Jaschob, R. Ramsden, T. N. Davis, E. G. Muller and A. Sali, *Determining protein complex structures based on a bayesian model of in Vivo Förster Resonance Energy Transfer (FRET) Data*, *Mol. Cell. Proteomics*, 2014, **13**, 2812–2823.

[29] A. T. Brunger, P. Strop, M. Vrljic, S. Chu and K. R. Weninger, *Three-dimensional molecular modeling with single molecule FRET*, *J. Struct. Biol.*, 2011, **173**, 497–505.

[30] Del45, *Wikimedia Commons*, `https://commons.wikimedia.org/wiki/File:Lysozym{_}diffraction.png`.

[31] CSIRO, *Wikimedia Commons*, `https://commons.wikimedia.org/wiki/File:CSIRO{_}ScienceImage{_}1304{_}Protein{_}crystals.jpg`.

[32] Vossman, *Wikimedia Commons*, `https://commons.wikimedia.org/wiki/File:Cryoem{_}groel.jpg`.

[33] L. Lutterotti, S. Matthies and H.-R. Wenk, *MAUD: a friendly Java program for material analysis using diffraction*, *IUCr Newsl. CPD*, 1999, **21**, 14–15.

[34] W. Wriggers, R. A. Milligan and J. A. McCammon, *Situs: A package for docking crystal structures into low-resolution maps from electron microscopy*, *J. Struct. Biol.*, 1999, **125**, 185–195.

[35] J. Meiler, *PROSHIFT: Protein chemical shift prediction using artificial neural networks*, *J. Biomol. NMR*, 2003, **26**, 25–37.

[36] B. Webb, K. Lasker, J. Velázquez-Muriel, D. Schneidman-Duhovny, R. Pellarin, M. Bonomi, C. Greenberg, B. Raveh, E. Tjioe, D. Russel and A. Sali, *Modeling of proteins and their assemblies with the integrative modeling platform*, *Methods Mol. Biol.*, 2014, **1091**, 277–295.

[37] B. Webb, S. Viswanath, M. Bonomi, R. Pellarin, C. H. Greenberg, D. Saltzberg and A. Sali, *Integrative structure modeling with the Integrative Modeling Platform*, *Protein Sci.*, 2018, **27**, 245–258.

[38] Z. Hall, A. Politis and C. V. Robinson, *Structural modeling of heteromeric protein complexes from disassembly pathways and ion mobility-mass spectrometry*, *Structure*, 2012, **20**, 1596–1609.

[39] A. Politis, A. Y. Park, Z. Hall, B. T. Ruotolo and C. V. Robinson, *Integrative modelling coupled with ion mobility mass spectrometry reveals structural features of the clamp loader in complex with single-stranded DNA binding protein*, *J. Mol. Biol.*, 2013, **425**, 4790–4801.

[40] R. Vernon, Y. Shen, D. Baker and O. F. Lange, *Improved chemical shift based fragment selection for CS-Rosetta using Rosetta3 fragment picker*, *J. Biomol. NMR*, 2013, **57**, 117–127.

[41] J. A. Kovacs, V. E. Galkin and W. Wriggers, *Accurate flexible refinement of atomic models against medium-resolution cryo-EM maps using damped dynamics 08 Information and Computing Sciences 0801 Artificial Intelligence and Image Processing*, *BMC Struct. Biol.*, 2018, **18**, 12.

[42] J. D. Eschweiler and A. T. Frank, *IMMS_modeler*, 2017, `https://github.com/jEschweiler/Urease`.

[43] A. Kulesza, E. G. Marklund, L. Macaleese, F. Chirot and P. Dugourd, *Bringing Molecular Dynamics and Ion-Mobility Spectrometry Closer Together: Shape Correlations, Structure-Based Predictors, and Dissociation*, *J. Phys. Chem. B*, 2018, **122**, 8317–8329.

[44] K. M. Andersson and S. Hovmöller, *The average atomic volume and density of proteins*, *Zeitschrift fur Krist. - New Cryst. Struct.*, 1998, **213**, 369–373.

[45] J. Tsai, R. Taylor, C. Chothia and M. Gerstein, *The packing density in proteins: Standard radii and volumes*, *J. Mol. Biol.*, 1999, **290**, 253–266.

[46] M. L. Quillin and B. W. Matthews, *Accurate calculation of the density of proteins*, *Acta Crystallogr. Sect. D Biol. Crystallogr.*, 2000, **56**, 791–794.

[47] P. G. Squire and M. E. Himmel, *Hydrodynamics and protein hydration*, *Arch. Biochem. Biophys.*, 1979, **196**, 165–177.

[48] K. Gekko and H. Noguchi, *Compressibility of globular proteins in water at 25C*, *J. Phys. Chem.*, 1979, **83**, 2706–2714.

[49] H. Fischer, I. Polikarpov and A. F. Craievich, *Average protein density is a molecular-weight-dependent function.*, *Protein Sci.*, 2004, **13**, 2825–8.

[50] A. Maißer, V. Premnath, A. Ghosh, T. A. Nguyen, M. Attoui and C. J. Hogan, *Determination of gas phase protein ion densities via ion mobility analysis with charge reduction*, *Phys. Chem. Chem. Phys.*, 2011, **13**, 21630–21641.

[51] E. G. Marklund, M. T. Degiacomi, C. V. Robinson, A. J. Baldwin and J. L. Benesch, *Collision cross sections for structural proteomics*, *Structure*, 2015, **23**, 791–799.

[52] P. Tan, J. Li and L. Hong, *Statistical properties for diffusive motion of hydration water on protein surface*, *Phys. B Condens. Matter*, 2019, **562**, 1–5.

[53] E. Zurlo, I. Gorroño Bikandi, N. J. Meeuwenoord, D. V. Filippov and M. Huber, *Tracking amyloid oligomerization with monomer resolution using a 13-amino acid peptide with a backbone-fixed spin label*, *Phys. Chem. Chem. Phys.*, 2019, **21**, 25187–25195.

[54] M. Ashoorirad, M. Saviz and A. Fallah, *On the electrical properties of collagen macromolecule solutions: Role of collagen-water interactions*, *J. Mol. Liq.*, 2020, **300**, 112344.

[55] A. Wargenau, N. Fekete, A. V. Beland, G. Sabbatier, O. M. Bowden, M. D. Boulanger and C. A. Hoesli, *Protein film formation on cell culture surfaces investigated by quartz crystal microbalance with dissipation monitoring and atomic force microscopy*, *Colloids Surfaces B Biointerfaces*, 2019, **183**, 110447.

[56] A. A. Ashkarran, K. S. Suslick and M. Mahmoudi, *Magnetically Levitated Plasma Proteins*, *Anal. Chem.*, 2020, **92**, 1663–1668.

[57] K. A. Dill, S. B. Ozkan, M. S. Shell and T. R. Weikl, *The Protein Folding Problem*, *Annu. Rev. Biophys.*, 2008, **37**, 289–316.

[58] S. A. Hollingsworth and R. O. Dror, *Molecular Dynamics Simulation for All*, *Neuron*, 2018, **99**, 1129–1143.

[59] N. S. Pagadala, K. Syed and J. Tuszynski, *Software for molecular docking: a review.*, *Biophys. Rev.*, 2017, **9**, 91–102.

[60] *Current Release Statistics < Uniprot < EMBL-EBI*, `https://www.ebi.ac.uk/uniprot/TrEMBLstats`.

[61] J. Moult, J. T. Pedersen, R. Judson and K. Fidelis, *A largescale experiment to assess protein structure prediction methods*, *Proteins Struct. Funct. Bioinforma.*, 1995, **23**, ii–iv.

[62] A. Kryshtafovych, T. Schwede, M. Topf, K. Fidelis and J. Moult, *Critical assessment of methods of protein structure prediction (CASP)Round XIII*, *Proteins Struct. Funct. Bioinforma.*, 2019, **87**, 1011–1020.

[63] K. T. Simons, C. Kooperberg, E. Huang and D. Baker, *Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions*, *J. Mol. Biol.*, 1997, **268**, 209–225.

[64] J. Yang, R. Yan, A. Roy, D. Xu, J. Poisson and Y. Zhang, *The I-TASSER suite: Protein structure and function prediction*, *Nat. Methods*, 2014, **12**, 7–8.

[65] D. Xu and Y. Zhang, *Toward optimal fragment generations for ab initio protein structure assembly*, *Proteins Struct. Funct. Bioinforma.*, 2013, **81**, 229–239.

[66] B. Webb and A. Sali, *Comparative protein structure modeling using MODELLER*, *Curr. Protoc. Bioinforma.*, 2016, **2016**, 5.6.1–5.6.37.

[67] A. Waterhouse, M. Bertoni, S. Bienert, G. Studer, G. Tauriello, R. Gumienny, F. T. Heer, T. A. De Beer, C. Rempfer, L. Bordoli, R. Lepore and T. Schwede, *SWISS-MODEL: Homology modelling of protein structures and complexes*, *Nucleic Acids Res.*, 2018, **46**, W296–W303.

[68] A. W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Žídek, A. W. R. Nelson, A. Bridgland, H. Penedones, S. Petersen, K. Simonyan, S. Crossan, P. Kohli, D. T. Jones, D. Silver, K. Kavukcuoglu and D. Hassabis, *Protein structure prediction using multiple deep neural networks in the 13th Critical Assessment of Protein Structure Prediction (CASP13)*, *Proteins Struct. Funct. Bioinforma.*, 2019, **87**, 1141–1148.

[69] A. W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Žídek, A. W. Nelson, A. Bridgland, H. Penedones, S. Petersen, K. Simonyan, S. Crossan,

P. Kohli, D. T. Jones, D. Silver, K. Kavukcuoglu and D. Hassabis, *Improved protein structure prediction using potentials from deep learning*, *Nature*, 2020, **577**, 706–710.

[70] A. Senior, R. Evans, J. Jumper, A. Pritzel, T. Green, M. Figurnov, K. Tunyasuvunakool, O. Ronneberger, R. Bates, A. Žídek, A. Bridgland, C. Meyer, S. A. A. Kohl, A. Potapenko, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, M. Steinegger, M. Pacholska, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli and D. Hassabis, *High Accuracy Protein Structure Prediction Using Deep Learning*, *Fourteenth Crit. Assess. Tech. Protein Struct. Predict.*, 2020.

[71] L. Casalino, Z. Gaieb, J. A. Goldsmith, C. K. Hjorth, A. C. Dommer, A. M. Harbison, C. A. Fogarty, E. P. Barros, B. C. Taylor, J. S. Mclellan, E. Fadda and R. E. Amaro, *Beyond shielding: The roles of glycans in the SARS-CoV-2 spike protein*, *ACS Cent. Sci.*, 2020, **6**, 1722–1734.

[72] O. Guvench and A. D. MacKerell, *Comparison of protein force fields for molecular dynamics simulations*, *Methods Mol. Biol.*, 2008, **443**, 63–88.

[73] P. Xu, E. B. Guidez, C. Bertoni and M. S. Gordon, *Perspective: Ab initio force field methods derived from quantum mechanics*, *J. Chem. Phys.*, 2018, **148**, 90901.

[74] P. Gkeka, G. Stoltz, A. Barati Farimani, Z. Belkacemi, M. Ceriotti, J. D. Chodera, A. R. Dinner, A. L. Ferguson, J. B. Maillet, H. Minoux, C. Peter, F. Pietrucci, A. Silveira, A. Tkatchenko, Z. Trstanova, R. Wiewiora and T. Lelièvre, *Machine Learning Force Fields and Coarse-Grained Variables in Molecular Dynamics: Application to Materials and Biological Systems*, *J. Chem. Theory Comput.*, 2020, **16**, 4757–4775.

[75] H. J. Berendsen, D. van der Spoel and R. van Drunen, *GROMACS: A message-passing parallel molecular dynamics implementation*, *Comput. Phys. Commun.*, 1995, **91**, 43–56.

[76] J. C. Phillips, D. J. Hardy, J. D. Maia, J. E. Stone, J. V. Ribeiro, R. C. Bernardi, R. Buch, G. Fiorin, J. Hénin, W. Jiang, R. McGreevy, M. C. Melo, B. K. Radak, R. D. Skeel, A. Singharoy, Y. Wang, B. Roux, A. Aksimentiev, Z. Luthey-Schulten, L. V. Kalé, K. Schulten, C. Chipot and E. Tajkhorshid, *Scalable molecular dynamics on CPU and GPU architectures with NAMD*, *J. Chem. Phys.*, 2020, **153**, 044130.

[77] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, P. A. Kollman and D. C. Spellmeyer, *A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules*, *J. Am. Chem. Soc.*, 1995, **117**, 5179–5197.

[78] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan and M. Karplus, *CHARMM: A program for macromolecular energy, minimization, and dynamics calculations*, *J. Comput. Chem.*, 1983, **4**, 187–217.

[79] W. F. Van Gunsteren and H. J. C. Berendsen, *The GROMOS Software for (Bio)Molecular Simulation GROMOS87 Groningen Molecular Simulation (GROMOS) Library Manual*, 1987, pp. 1–221.

[80] C. G. Don and M. Smieško, *Microsecond MD simulations of human CYP2D6 wild-type and five allelic variants reveal mechanistic insights on the function*, *PLoS One*, 2018, **13**, e0202534.

[81] J. D. Durrant and J. A. McCammon, *Molecular dynamics simulations and drug discovery*, *BMC Biol.*, 2011, **9**, 71.

[82] D. W. Li and R. Brüschweiler, *Protocol to make protein NMR structures amenable to stable long time scale molecular dynamics simulations*, *J. Chem. Theory Comput.*, 2014, **10**, 1781–1787.

[83] P. Herrera-Nieto, A. Pérez and G. De Fabritiis, *Characterization of partially ordered states in the intrinsically disordered N-terminal domain of p53 using millisecond molecular dynamics simulations*, *Sci. Rep.*, 2020, **10**, 12402.

[84] K. F. O'Rourke, J. M. Axe, R. N. D'Amico, D. Sahu and D. D. Boehr, *Millisecond Timescale Motions Connect Amino Acid Interaction Networks in Alpha Tryptophan Synthase*, *Front. Mol. Biosci.*, 2018, **5**, 92.

[85] A. Zhuravleva and D. M. Korzhnev, *Protein folding by NMR*, *Prog. Nucl. Magn. Reson. Spectrosc.*, 2017, **100**, 52–77.

[86] P. F. Almeida, W. L. Vaz and T. E. Thompson, *Lipid diffusion, free area, and molecular dynamics simulations*, *Biophys. J.*, 2005, **88**, 4434–4438.

[87] S. J. Marrink, A. H. De Vries and A. E. Mark, *Coarse Grained Model for Semiquantitative Lipid Simulations*, *J. Phys. Chem. B*, 2004, **108**, 750–760.

[88] A. M. Fluitt and J. J. De Pablo, *An Analysis of Biomolecular Force Fields for Simulations of Polyglutamine in Solution*, *Biophys. J.*, 2015, **109**, 1009–1018.

[89] M. Nemec and D. Hoffmann, *Quantitative Assessment of Molecular Dynamics Sampling for Flexible Systems*, *J. Chem. Theory Comput.*, 2017, **13**, 400–414.

[90] Y. I. Yang, Q. Shao, J. Zhang, L. Yang and Y. Q. Gao, *Enhanced sampling in molecular dynamics*, *J. Chem. Phys.*, 2019, **151**, 70902.

[91] M. T. Degiacomi, *Coupling Molecular Dynamics and Deep Learning to Mine Protein Conformational Space*, *Structure*, 2019, **27**, 1034–1040.e3.

[92] Z. Liu, A. Szarecka, M. Yonkunas, K. Speranskiy, M. Kurnikova and M. Cascio, *Crosslinking constraints and computational models as complementary tools in modeling the extracellular domain of the glycine receptor*, *PLoS One*, 2014, **9**, e102571.

[93] S. E. Winograd-Katz, R. Fässler, B. Geiger and K. R. Legate, *The integrin adhesome: from genes and proteins to human disease.*, *Nat. Rev. Mol. Cell Biol.*, 2014, **15**, 273–288.

[94] Y. Shan, E. T. Kim, M. P. Eastwood, R. O. Dror, M. A. Seeliger and D. E. Shaw, *How does a drug molecule find its target binding site?*, *J. Am. Chem. Soc.*, 2011, **133**, 9181–9183.

[95] J. Janin, *Assessing predictions of protein-protein interaction: The CAPRI experiment*, *Protein Sci.*, 2005, **14**, 278–283.

[96] R. Chen, L. Li and Z. Weng, *ZDOCK: An initial-stage protein-docking algorithm*, *Proteins Struct. Funct. Genet.*, 2003, **52**, 80–87.

[97] A. Tovchigrechko and I. A. Vakser, *GRAMM-X public web server for protein-protein docking*, *Nucleic Acids Res.*, 2006, **34**, W310–W314.

[98] D. Kozakov, R. Brenke, S. R. Comeau and S. Vajda, *PIPER: An FFT-based protein docking program with pairwise potentials*, *Proteins Struct. Funct. Bioinforma.*, 2006, **65**, 392–406.

[99] G. R. Smith, M. J. Sternberg and P. A. Bates, *The Relationship between the Flexibility of Proteins and their Conformational States on Forming ProteinProtein Complexes with an Application to ProteinProtein Docking*, *J. Mol. Biol.*, 2005, **347**, 1077–1101.

[100] R. Grünberg, J. Leckner and M. Nilges, *Complementarity of Structure Ensembles in Protein-Protein Binding*, *Structure*, 2004, **12**, 2125–2136.

[101] M. Król, A. L. Tournier and P. A. Bates, *Flexible relaxation of rigid-body docking solutions*, *Proteins Struct. Funct. Bioinforma.*, 2007, **68**, 159–169.

[102] R. M. Jackson, H. A. Gabb and M. J. Sternberg, *Rapid refinement of protein interfaces incorporating solvation: application to the docking problem*, *J. Mol. Biol.*, 1998, **276**, 265–285.

[103] M. Król, R. A. Chaleil, A. L. Tournier and P. A. Bates, *Implicit flexibility in protein docking: Cross-docking and local refinement*, *Proteins Struct. Funct. Bioinforma.*, 2007, **69**, 750–757.

[104] J. J. Gray, S. Moughon, C. Wang, O. Schueler-Furman, B. Kuhlman, C. A. Rohl and D. Baker, *Protein-protein docking with simultaneous optimization of rigid-body displacement and side-chain conformations*, *J. Mol. Biol.*, 2003, **331**, 281–299.

[105] M. Zacharias, *Protein-protein docking with a reduced protein model accounting for side-chain flexibility*, *Protein Sci.*, 2003, **12**, 1271–1282.

[106] M. Zacharias, Proteins Struct. Funct. Genet., 2005, pp. 252–256.

[107] D. Schneidman-Duhovny, Y. Inbar, R. Nussinov and H. J. Wolfson, Proteins Struct. Funct. Genet., 2005, pp. 224–231.

[108] V. I. Lesk and M. J. Sternberg, *3D-Garden: A system for modelling protein-protein complexes based on conformational refinement of ensembles generated with the marching cubes algorithm*, *Bioinformatics*, 2008, **24**, 1137–1144.

[109] I. H. Moal and P. A. Bates, *SwarmDock and the use of normal modes in protein-protein Docking*, *Int. J. Mol. Sci.*, 2010, **11**, 3623–3648.

[110] J. Fernández-Recio, M. Totrov and R. Abagyan, *ICM-DISCO docking by global energy optimization with fully flexible side-chains*, *Proteins Struct. Funct. Bioinforma.*, 2003, **52**, 113–117.

[111] C. Dominguez, R. Boelens and A. M. Bonvin, *HADDOCK: A protein-protein docking approach based on biochemical or biophysical information*, *J. Am. Chem. Soc.*, 2003, **125**, 1731–1737.

[112] D. Russel, K. Lasker, B. Webb, J. Velázquez-Muriel, E. Tjioe, D. Schneidman-Duhovny, B. Peterson and A. Sali, *Putting the pieces together: Integrative modeling platform software for structure determination of macromolecular assemblies*, *PLoS Biol.*, 2012, **10**, e1001244.

[113] S. Hauri, H. Khakzad, L. Happonen, J. Teleman, J. Malmström and L. Malmström, *Rapid determination of quaternary protein structures in complex biological samples*, *Nat. Commun.*, 2019, **10**, 1–10.

[114] T. Vreven, I. H. Moal, A. Vangone, B. G. Pierce, P. L. Kastritis, M. Torchala, R. Chaleil, B. Jiménez-García, P. A. Bates, J. Fernandez-Recio, A. M. J. J. Bonvin and Z. Weng, *Updates to the Integrated Protein-Protein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2*, *J. Mol. Biol.*, 2015, **427**, 3031–3041.

[115] R. F. Alford, J. Koehler Leman, B. D. Weitzner, A. M. Duran, D. C. Tilley, A. Elazar and J. J. Gray, *An Integrated Framework Advancing Membrane Protein Modeling and Design*, *PLoS Comput. Biol.*, 2015, **11**, e1004398.

[116] N. Hurwitz, D. Schneidman-Duhovny and H. J. Wolfson, *Memdock: An α-helical membrane protein docking algorithm*, *Bioinformatics*, 2016, **32**, 2444–2450.

[117] A. Tovchigrechko and I. A. Vakser, Proteins Struct. Funct. Genet., 2005, pp. 296–301.

[118] S. Viswanath, L. Dominguez, L. S. Foster, J. E. Straub and R. Elber, *Extension of a protein docking algorithm to membranes and applications to amyloid precursor protein dimerization*, *Proteins Struct. Funct. Bioinforma.*, 2015, **83**, 2170–2185.

[119] B. Pierce and Z. Weng, *ZRANK: Reranking protein docking predictions with an optimized energy function*, *Proteins Struct. Funct. Genet.*, 2007, **67**, 1078–1086.

[120] D. Kozakov, D. R. Hall, D. Beglov, R. Brenke, S. R. Comeau, Y. Shen, K. Li, J. Zheng, P. Vakili, I. C. Paschalidis and S. Vajda, *Achieving reliability and high accuracy in*

automated protein docking: ClusPro, PIPER, SDU, and stability analysis in CAPRI rounds 13-19, *Proteins Struct. Funct. Bioinforma.*, 2010, **78**, 3124–3130.

[121] P. I. Koukos, I. Faro, C. W. van Noort and A. M. Bonvin, *A Membrane Protein Complex Docking Benchmark*, *J. Mol. Biol.*, 2018, **430**, 5246–5256.

[122] B. Jiménez-García, J. Roel-Touris, M. Romero-Durana, M. Vidal, D. Jiménez-González and J. Fernández-Recio, *LightDock: a new multi-scale approach to proteinprotein docking*, *Bioinformatics*, 2018, **34**, 49–55.

[123] J. Roel-Touris, B. Jiménez-García and A. Bonvin, *Integrative Modeling of Membrane-associated Protein Assemblies*, *Nat. Commun.*, 2020, **11**, 6210.

[124] D. A. Pearlman, D. A. Case, J. W. Caldwell, W. S. Ross, T. E. Cheatham, S. DeBolt, D. Ferguson, G. Seibel and P. Kollman, *AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules*, *Comput. Phys. Commun.*, 1995, **91**, 1–41.

[125] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, *ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB*, *J. Chem. Theory Comput.*, 2015, **11**, 3696–3713.

[126] J. Huang and A. D. Mackerell, *CHARMM36 all-atom additive protein force field: Validation based on comparison to NMR data*, *J. Comput. Chem.*, 2013, **34**, 2135–2145.

[127] B. Hess, H. Bekker, H. J. Berendsen and J. G. Fraaije, *LINCS: A Linear Constraint Solver for molecular simulations*, *J. Comput. Chem.*, 1997, **18**, 1463–1472.

[128] G. Lamoureux, A. D. MacKerell and B. Roux, *A simple polarizable model of water based on classical drude oscillators*, *J. Chem. Phys.*, 2003, **119**, 5185–5197.

[129] T. P. Senftle, S. Hong, M. M. Islam, S. B. Kylasa, Y. Zheng, Y. K. Shin, C. Junkermeier, R. Engel-Herbert, M. J. Janik, H. M. Aktulga, T. Verstraelen, A. Grama and A. C. Van Duin, *The ReaxFF reactive force-field: Development, applications and future directions*, 2016, `http://www.nature.com/articles/npjcompumats201511`.

[130] D. Petrović, X. Wang and B. Strodel, *How accurately do force fields represent protein side chain ensembles?*, *Proteins Struct. Funct. Bioinforma.*, 2018, **86**, 935–944.

[131] R. Eberhart and J. Kennedy, Proc. Int. Symp. Micro Mach. Hum. Sci., 1995, pp. 39–43.

[132] A. Shrake and J. A. Rupley, *Environment and exposure to solvent of protein atoms. Lysozyme and insulin*, *J. Mol. Biol.*, 1973, **79**, 351–71.

[133] J. Weiser, P. S. Shenkin and W. C. Still, *Approximate atomic surfaces from linear combinations of pairwise overlaps (LCPO)*, *J. Comput. Chem.*, 1999, **20**, 217–230.

[134] K. V. Klenin, F. Tristram, T. Strunk and W. Wenzel, *Derivatives of molecular surface area and volume: Simple and exact analytical formulas*, *J. Comput. Chem.*, 2011, **32**, 2647–2653.

2 A novel representation of molecular structure and dynamics

2.1 Introduction

Treating a protein, *in silico*, as some coarse-grained representation to bypass side-chain flexibility is not a novel concept.[1] The critical issue with previous approaches that adopt this tactic,[2–5] however, is that fundamentally they build their volumetric representations from a fixed set of atomic positions, and, in the case of docking, it is only during a post-processing step that some flexible refinement is considered. Simultaneous consideration of a protein's dynamic nature while building a volumetric representation seems the natural progression from these methods, given the widespread ease and speed at which Molecular Dynamics calculations can be made, in contrast to the mid-1990s to late-2000s when many of these methods were developed.

In this Chapter, we shall discuss how we can use Molecular Dynamics as a foundation to build a protein model capable of representing protein shape, electrostatics and side-chain dynamics, for application in a variety of common biophysical problems. This work's eventual goal is to ensure that our representation, known as Spatial and Temporal Influence Density (STID) map, is sufficient for protein-protein docking purposes. We shall begin with the theory behind our representation, as well as the objective benchmarking applied both during the construction phase, and later to check that the resultant shape is a good, physical model of a protein, and not just some abstract object solely applicable for a single purpose.[6] Given the success in demonstrating that our representation is related to some protein characteristics (Section 2.3.4), we then move on to establish its value in a host of other biophysical problems. Specifically, to measure the variation in the permittivity of a system on a nanometre scale, to predict the density of a protein and provide a new dynamics-based dependency, and to give insight into allosteric changes at a residue level following some binding event.

The dielectric work discussed in Section 2.4 was partially successful, as our maps were able to predict the general trend for a set of solvent's dielectric constants. The STID map's ability to estimate the known correlation between this critical, physical, quantity and polarisability acts as a validation that our model is not just some abstract concept. Thus, we avoid falling into the common trap for these shape-based approaches to protein modelling.

In Section 2.5 we show that STID maps, which are essentially excluded volumes at the cutoff we derive in Section 2.3.4, are capable of predicting protein densities in different environments. Specifically, we generate maps in both a water solvent and *in vacuo*, matching experimental data for both. This leads to a new definition for determining the density of protein, a topic that has remained one of dispute since the late 1990s. Rather than a molecular weight dependency as previous works have indicated,[7] the densities calculated through our STID map based measurements, validated through experimental data, indicate a dynamics based relationship.

Finally, in Section 2.6 we show that by comparing STID maps of individual residues extracted from different bound complexes, we can highlight which residues have their local conformational dynamics altered due to some binding event. While it does not elucidate the allosteric process itself, nor any large conformational changes, it does provide a good starting point for understanding which distal (and proximal) residues are intimately connected, and subsequently impacted as a result of some interaction with a binding partner.

2.2 Theory: building the Spatial and Temporal Influence Density map

The following theory is built upon the principle that a volumetric map can consider both the structural and dynamic properties of a protein. It is constructed from the inherent motion of charged atoms *via* the time-averaged dipoles forming within a localised space. The complete methodology, represented in Figure 2.2.1, is summarised as follows:

1. The PDB file of a protein structure is parameterized according to a desired atomistic force field.
2. The protein is immersed in a water box with Na^+ or Cl^- acting as counter-ions and automatically subjected to energy minimisation, followed by an MD equilibration and production protocol.
3. A dipole map, following the Onsager theory of dielectric polarisation, is derived from the produced MD trajectory.
4. The dipole map is converted into a 3-dimensional grid of points, each containing a pseudo-atom with a characteristic van der Waals radius.
5. Each pseudo-atom is used to define a local Gaussian distribution, with a shape determined by the van der Waals radius of the pseudo-atom.
6. A volumetric map is produced by summing, on each grid point, the value of local and neighbouring Gaussians. The resulting map is finally normalised between 0 and 1.

We only consider the contribution of protein atoms, including hydrogen atoms, when constructing a STID map. In other words, we ignore the presence of water, salts, lipids or other non-protein molecules. Despite their absence, the impact of these non-protein atoms in the MD simulation will be reflected in the STID map through changes to how side-chains explore their local conformational space, and subsequently, their position uncertainty.

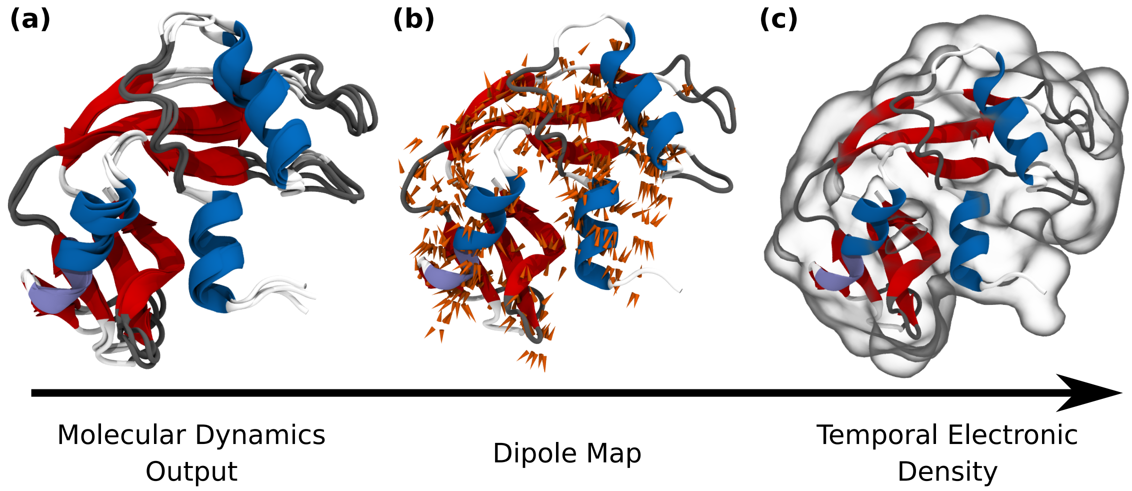

Figure 2.2.1: Pipeline for the generation of STID maps. **(a)** Super-imposition of multiple structures from the MD simulation of ribonuclease A (PDB: 9RSA), coloured by secondary structure (α-helices as blue, 3_{10}-helices as light purple, β-sheets as red, unstructured coils as white, and turns as grey). **(b)** Superimposed dipole map generated from the simulation. For clarity, only dipoles greater than 0.8 D are shown, D referring to the CGS unit of Debye, commonly deployed to represent electric dipole moments (1 D = 3.33564×10^{-30} C m in SI units). **(c)** Final STID map derived from the dipole map, represented by an isosurface through the volumetric grid.

2.2.1 From protein structure to dipole map

The fast local rearrangements of side-chain motion are sampled through a short MD simulation. Consistent with experimental NMR evidence, we have found that 600 ps is enough for this purpose (see benchmark and Figures 2.3.4 & 2.3.5 in Section 2.3.2). We align the resulting protein trajectory according to the centre of mass of the molecule, and arrange it within a stationary, cubic grid, wherein each voxel is 1 Å across in x, y and z (a parameter determined quantitatively in a benchmark discussed in Section 2.3.1). We use this information to calculate local dipoles on each grid point. To this end, we are expanding upon the theory laid out by Kirkwood,[8] Fröhlich[9] and Neumann *et al.*,[10] which describe the fundamental theory of dielectrics.

Following the Onsager theory of dielectric polarisation, we represent each voxel, v, as a spherical solute with volume, V_{v}, with an internal permittivity, ϵ_{v}, embedded within a uniform dielectric continuum with permittivity ϵ_{Ex}. The charge distribution inside the voxel is that of several point charges, with a dipole associated with the centre. Point charges within the neighbouring voxels on each Cartesian edge and corner (*i.e.*, a total of 26 neighbours, a quantity determined in a benchmark shown in Figures 2.3.3 & 2.3.7 in Section 2.3.1) are also associated with the central voxel. A sliding window is applied spatially such that a point charge at a time, t, will contribute to 27 different voxels in total. A visual guide for how the sliding window is applied is provided in Section 2.3.1's benchmark through Figure 2.3.2. Given the fluctuations of the dipole moment of the solute, $\overrightarrow{M}_{\mathrm{v}}$, observed over the simulation in a voxel, it is possible to calculate ϵ_{v}.

The Fröhlich-Kirkwood model states that ϵ_{v} is a function of the total dipole's second moment probability distribution, with $\overrightarrow{M}_{\mathrm{v}}$ given by:

$$\overrightarrow{M}_{\mathrm{v}} = \sum_{i=0}^{N} q_{i,\mathrm{v}} \overrightarrow{r}_{i,\mathrm{v}}, \tag{2.2.1}$$

where $q_{i,\mathrm{v}}$ is the charge of atom i at distance $\overrightarrow{r}_{i,\mathrm{v}}$ from the geometrical centre of voxel v. N is the number of atoms that contribute to a voxel's dipole moment. The charges are obtained from the forcefield used for the simulation. For a solute with a net charge, which all voxels with atoms passing through will have, $\overrightarrow{M}_{\mathrm{v}}$ is dependent on the origin; thus the grid is fixed in time and space. Therefore, we can produce a dipole map delivering a representation of local vectorial electrostatic characteristics of a region of space occupied by a molecule (see Figures 2.2.1(b) & 2.3.8).

2.2.2 From dipole map to Spatial and Temporal Influence Density map

We can now leverage on the obtained dipole map to derive volumetric information on a molecule, *i.e.* a quantity that is easier to visualise and use in a variety of biophysical contexts, though primarily protein-protein docking. To this end, it is necessary to convert our dipolar vectorial representation into a scalar quantity. Under the Fröhlich–Kirkwood model, we can relate the fluctuations of each dipole moment, $\overrightarrow{M}_{\mathrm{v}}$, to the voxel's dielectric, ϵ_{v} by

$$\frac{\langle \overrightarrow{M}_{\mathrm{v}}^2 \rangle - \langle \overrightarrow{M}_{\mathrm{v}} \rangle^2}{3\epsilon_0 V_{\mathrm{v}} k_{\mathrm{B}} T_{\mathrm{v}}} = \frac{(2\epsilon_{\mathrm{Ex}} + 1)(\epsilon_{\mathrm{v}} - 1)}{2\epsilon_{\mathrm{Ex}} + \epsilon_{\mathrm{v}}}, \tag{2.2.2}$$

where the pointed brackets denotes the average, the square of $\overrightarrow{M}_{\mathrm{v}}$ refers to the dot product, k_{B} is the Boltzmann constant (1.3806×10^{-23} m^2 kg s^{-2} K^{-1}), ϵ_0 is the permittivity of free space (8.8542×10^{-12} m^{-3} kg^{-1} s^4 A^2), T_{v} is the internal temperature of a voxel, approximated as the temperature of the system, V_{v} the volume of a voxel, and ϵ_{Ex} is set to 54 following work by Pitera *et al.*[11] . Solving this for ϵ_{v} gives

$$\epsilon_{\mathrm{v}} = \frac{1 + \frac{\langle \overrightarrow{M}_{\mathrm{v}}^2 \rangle - \langle \overrightarrow{M}_{\mathrm{v}} \rangle^2}{3\epsilon_0 V_{\mathrm{v}} k_{\mathrm{B}} T_{\mathrm{v}}} \frac{2\epsilon_{\mathrm{Ex}}}{2\epsilon_{\mathrm{Ex}}+1}}{1 - \frac{\langle \overrightarrow{M}_{\mathrm{v}}^2 \rangle - \langle \overrightarrow{M}_{\mathrm{v}} \rangle^2}{3\epsilon_0 V_{\mathrm{v}} k_{\mathrm{B}} T_{\mathrm{v}}} \frac{1}{2\epsilon_{\mathrm{Ex}}+1}}. \tag{2.2.3}$$

Each ϵ_{v} value derived from the dipole map now encodes information on the local dynamics and atomic charges. Our next step is to convert the resulting dielectric map into a quantity that relates to a pseudo-electron density. To do so, we place a pseudo-atom at the centre of each voxel and calculate its polarisability, α_{v}, using the Clausius—Mossotti equation:

$$\alpha_{\mathrm{v}} = \frac{3\epsilon_0}{N_{\mathrm{v}}} \left(\frac{\epsilon_{\mathrm{v}} - 1}{\epsilon_{\mathrm{v}} + 2} \right), \tag{2.2.4}$$

where N_{v} is the number density inside the voxel. Since N_{v} is derived from the number of pseudo-atoms inside a voxel, by definition one, we can simply set it as the inverse of V_{v}. Fedorov *et al.*[12] showed that α_{v} can be related to a van der Waals radius, R_{vdW}, through a scaling relationship based on a quantum Drude oscillator model:

$$R_{\mathrm{vdW}} = 2.54\alpha_{\mathrm{v}}^{(1/7)}, \tag{2.2.5}$$

where the constant 2.54 is a universal scaling factor between electron density and atomic volume at R_{vdW} in atomic units. While Fedorov *et al.* note that a full derivation of this constant is still incomplete, they demonstrate that the relationship in Equation 2.2.5 gives theoretical quantities closer to experimental data than previous models based on classical hard-sphere representations.

The electrostatic and dynamic information encapsulated in the local R_{vdW} values is now suitable to be transformed into a quantity encoding a pseudo-electron density. To this end, we assume that each pseudo-atom radius is equal to the full width at half maximum of a decaying function, here defined as a 3-dimensional Gaussian, with the maximum at the voxel's centre. This allows each pseudo-electron density to "leak" into neighbouring voxels, which is reasonable given that a central voxel's behaviour is characterised by the atoms in its neighbourhood.

Contributions from any Gaussians with a non-zero value present within a voxel are then summed, and the resulting map is finally normalised by dividing the values within each voxel across the overall map by the maximum value found, *i.e.* the voxel containing the largest numerical value from the various Gaussian contributions will now contain a value of 1. An isosurface example is shown in Figure 2.2.1. These subsequent molecular representations we call Spatial and Temporal Influence Density (STID) maps. Because of this methodology, regions inside the protein's conformational space that are visited often or are highly charged will have greater associated STID values. This property makes STID maps a useful tool in a variety of applications as we will discuss. Most notably, this representation is very profitable in a protein-protein docking scenario (see Chapter 3), with the electrostatics arising from rapid side-chain motions, often ignored by other docking software, now accounted for.

2.3 Benchmarking and structural characterisation

The primary factors that impact the topological features of a STID map are the size of the voxels, the number of neighbouring voxels we consider when determining R_{vdW}, and the time over which we generate the map. Here, we will consider each of these in turn, before finishing with the derivation of a physical relationship between our STID maps and the structure of a protein.

2.3.1 STID map parameter determination

The voxel size and number of neighbours play an important role in determining the resolution and weightings we give to different atomic species in a STID map's topology. Consequently, our choice for these values has significant implications for future applications. We, therefore, performed a series of benchmarks to determine the best quantities for these parameters objectively. Our protein dataset is composed of a diverse set of 118 water-soluble proteins from the CAPRI benchmark[13] with various sizes, shapes and secondary structure. These cases display a range of inherent flexibilities, and as they are extracted from a protein docking benchmark, are subsequently classified according to how challenging their flexibility makes predictive docking. The specifics on what the three classifications (easy, medium and difficult) relate to, are given in Section 3.2.3.1.

Every region of space has its local STID characteristics determined based on the dynamics of atoms in a local neighbourhood of a pre-determined size. We define a STID map's resolution as the dimensions in Ångströms of this neighbourhood, *i.e.*, the diameter of a sphere centred on a point of interest. As here space is voxelated, the resolution is approximated as the edge length of a cubic cell centred around a point of interest. Two parameters are associated with a STID map's resolution: the voxel's size and the number of continuous voxels accounted for when sampling a specific region of space, or window size. We need to find a suitable resolution, too small and we introduce noise through pockets of vacuum with a large number of empty voxels, too large and we ignore residue-level detail. We consider STID maps with a higher average STID value, $D_{\mathrm{avg.}}$, to contain the most useful structural information because this would minimise low occupied or empty voxels, which contribute little. Therefore, resolution parameters were chosen to maximise the $D_{\mathrm{avg.}}$ value over all the proteins tested.

We first performed a sweep of possible voxel sizes (keeping the window size at 1), and measured their influence on $D_{\mathrm{avg.}}$ (Figure 2.3.1). In this test, we altered the voxel size while ignoring contributions from surrounding voxels. Results show that maximum $D_{\mathrm{avg.}}$ is obtained, on average, with 3 Å-wide voxels. The light blue band in

Figure 2.3.1(a) indicates one standard deviation for the mean $D_{\text{avg.}}$ across the dataset. Contrastingly, in Figure 2.3.1(b), we show the average of every individual $D_{\text{avg.}}$ standard deviation for each benchmarked protein, σ_{Points}. Between 3 Å and 6 Å, σ_{Points} remains roughly constant at its peak. Thus, in this region, we would expect to observe the greatest range of structural detail encompassed within a STID map for any specific protein. Thus, considering this range coupled with the peak of the mean of $D_{\text{avg.}}$ for the dataset, we conclude that a resolution of 3 Å is most appropriate to extract the greatest range of information from a STID map.

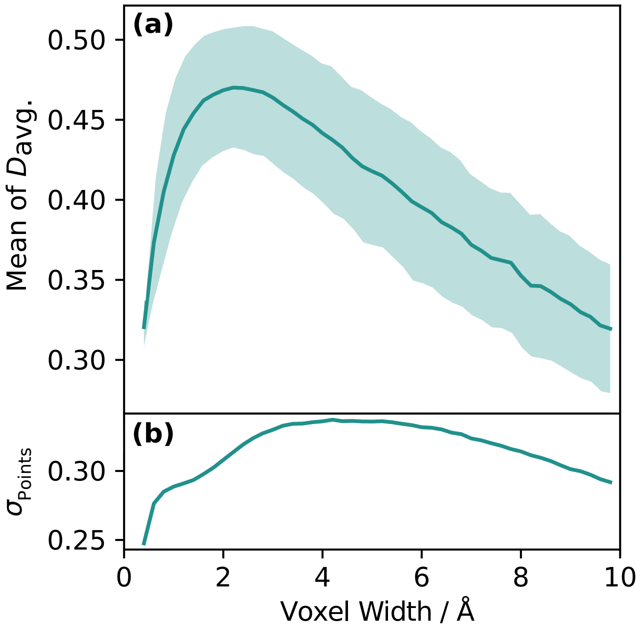

Figure 2.3.1: **(a)** The mean of $D_{\text{avg.}}$ across 118 benchmarked proteins with increasing voxel width. The blue band indicates one standard deviation. The shape of the line indicates that the overall resolution should be chosen between 1.5 Å and 3 Å. **(b)** The mean standard deviation of $D_{\text{avg.}}$ across the 118 proteins with increasing voxel width. σ_{Points} maintaining the same value through 3 Å to 6 Å demonstrates that the overall resolution of the STID maps should be within this region.

To determine the number of neighbouring voxels to include within the calculation of the STID value in a central voxel, the window size, we explored the effects of increasing the number of nearest neighbours with voxel width. Figure 2.3.2 gives a visual guide for how the sliding window is applied with an increasing number of neighbours considered. Figure 2.3.3 shows the impact the number of nearest neighbours has on $D_{\text{avg.}}$ for different window size, illustrated by the differing colours. The dark blue line represents the case where contributions from neighbouring voxels are ignored; only the central voxel itself is used, and it thus possesses a window size of one.

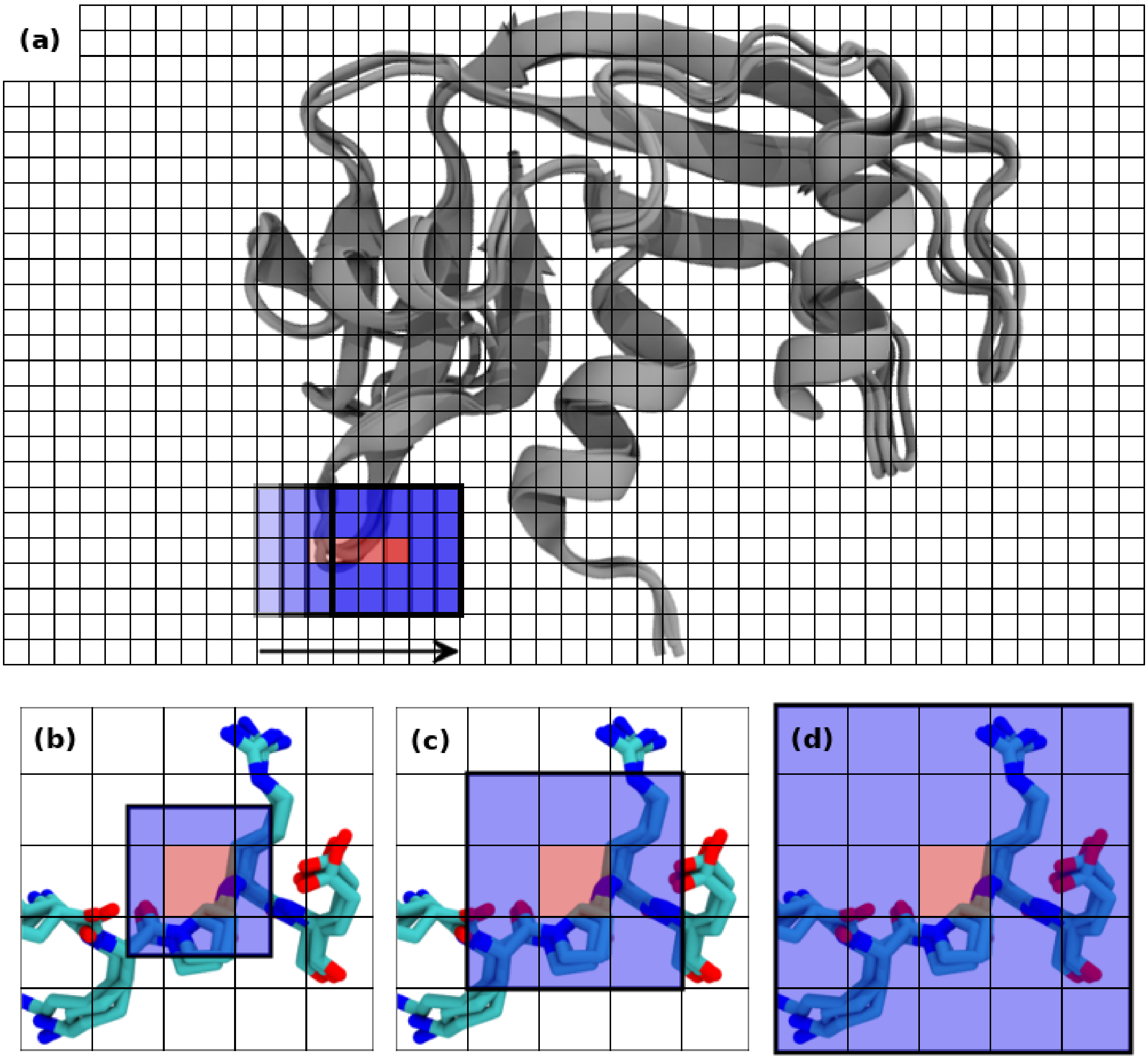

Figure 2.3.2: **(a)** A voxel grid extended over the conformational space explored by ribonuclease A (PDB: 9RSA). The grid also extends back in 3D. The sliding window (highlighted by the thicker border), demonstrates how the atomic motion in neighbouring voxels (in blue) contribute to the central voxel (red). The transparency of the box indicates which window it belongs to, with the "newer" windows darker. **(b)** A focus on this sliding window with 2 voxels contributing to the properties of the central red voxel. **(c)** Number of voxels is 3. **(d)** Number of voxels is 5.

In Figure 2.3.3, all cases peak at a $D_{\text{avg.}}$ = ~0.57, with the exception of the single voxel window case. Given this lack of consistency, and our consideration that a larger $D_{\text{avg.}}$ indicates a richer breadth of information; it is not appropriate to choose a window size of 1 voxel. Using a window size of two voxels to define the local neighbourhood (*i.e.*, a central voxel, and half a voxel in each direction) leads to $D_{\text{avg.}}$ peaking at a voxel width of 1.5 Å. With a window size of three (one central voxel, and one additional voxel along each edge and corner) $D_{\text{avg.}}$ peaks at a voxel width of 1 Å. This pattern repeats with larger window sizes. In other words, $D_{\text{avg.}}$ is maximised when the voxel width multiplied by the window size is 3 Å, consistent with the results shown in Figure 2.3.1. Thus, to obtain our optimal resolution of 3 Å; a voxel width of 1 Å and a window size of 3 should be used when producing the STID maps. While using two voxels, each having a size of 1.5 Å, could have also been a viable option, we preferred to use three neighbouring voxels as this simplifies the calculation of the Gaussian distributions that define the pseudo-electron density. Any smaller voxel widths would have a greater demand for computational resources.

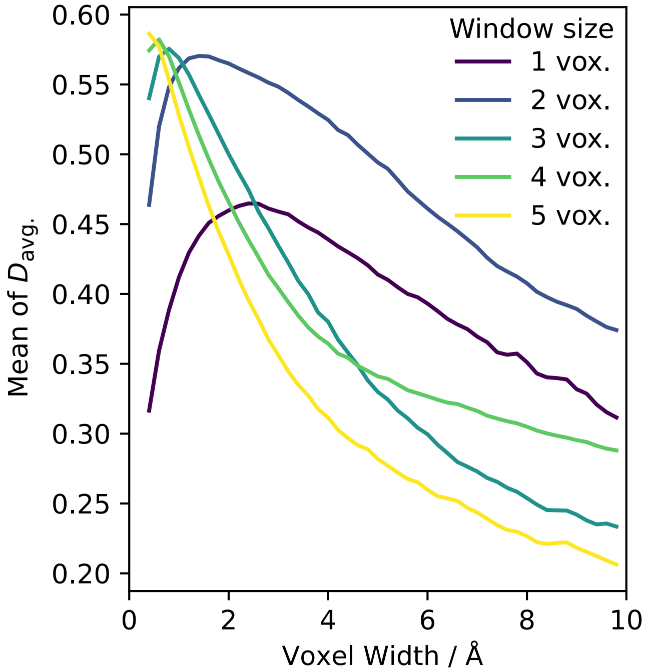

Figure 2.3.3: The impact on $D_{\text{avg.}}$ of increasing the window size with voxel width.

2.3.2 Convergence of side-chain motion

To ensure that a STID map is truly representative of side-chain motion and not simply a snapshot of it, we monitored how the values inside the voxels of the STID maps evolved over time. We calculated the Pearson cross-correlation coefficient (CCC) between successive STID maps, built from increasingly long molecular dynamics trajectories. The CCC, in this context, provides a measure of the similarity between two identically sized STID maps, where we compare the STID value in specific gridpoints with their counterpart point in another STID map. A CCC value approaching 1 means the maps are identical, while −1 indicates the opposite. Figure 2.3.4 illustrates this, with the CCC increasing rapidly over a very short period, before converging to 1. The CCC reported is only between STID map's non-zero values. While the exact time of convergence to the arbitrarily high CCC value of 0.9997 varies with protein, the Gaussian fitted in Figure 2.3.4(b) shows that, on average, CCCs converge in 197 ps, and 98% have converged within 300 ps. This timescale agrees with NMR experimental data, with the consensus that side-chain motion occurs on a picosecond timescale.[14] Quantitatively, using ^{13}C-multiplet NMR Mikhailov *et al.*[15] found that GLY-LYS-GLY side-chain cross-correlation times were on the order of 120 ± 10 ps at pH 6, which is similar to the timescales observed here.

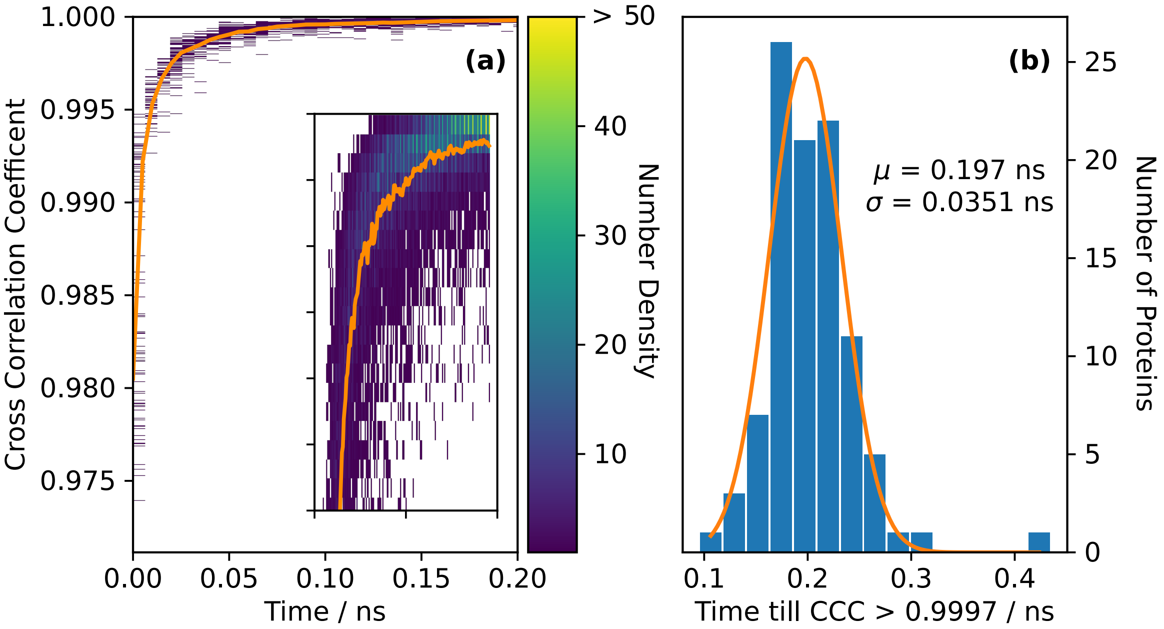

Figure 2.3.4: (a) Cross-Correlation Coefficient (CCC) between successive STID maps built up with increasingly long timescales. The orange line indicates an average across 118 proteins; the colourmap represents the number density of proteins at each CCC for each stretch of time. The inset is a zoom into the top 1% of the figure. The CCC is calculated between non-zero values. **(b)** The number of proteins achieving a CCC of 0.9997 by the listed time. A Gaussian (orange) has been fitted to the histogram. All protein motion converged within 500 ps, with the majority in the first 300 ps. The mean (μ) and standard deviation (σ) are reported.

To ensure that this convergence (Figure 2.3.4) was primarily due to side-chain motion and not because of any larger backbone motions, we calculated the root-mean-square-fluctuation (RMSF) for the Cα atoms for all proteins in the dataset and categorised their distribution by their associated secondary structure (sheet, helix and coil, see Figure 2.3.5). Results show that:

1. Fluctuations follow the expected trend, with coils exhibiting a greater RMSF than helices and sheets.
2. Known more flexible proteins tend to display larger overall RMSF.
3. RMSFs are, in the vast majority of cases, under 1 Å (less than the resolution of 3 Å used to calculate a voxel's STID value), thus indicating that the dynamics contributing to our STID maps are primarily due to the side-chain motion.

These results are consistent with the observation that the correlation times of fast, flexible loops are expected to be of the order of ~10 ns,[16] over one order of magnitude slower than the simulation timescales used here.

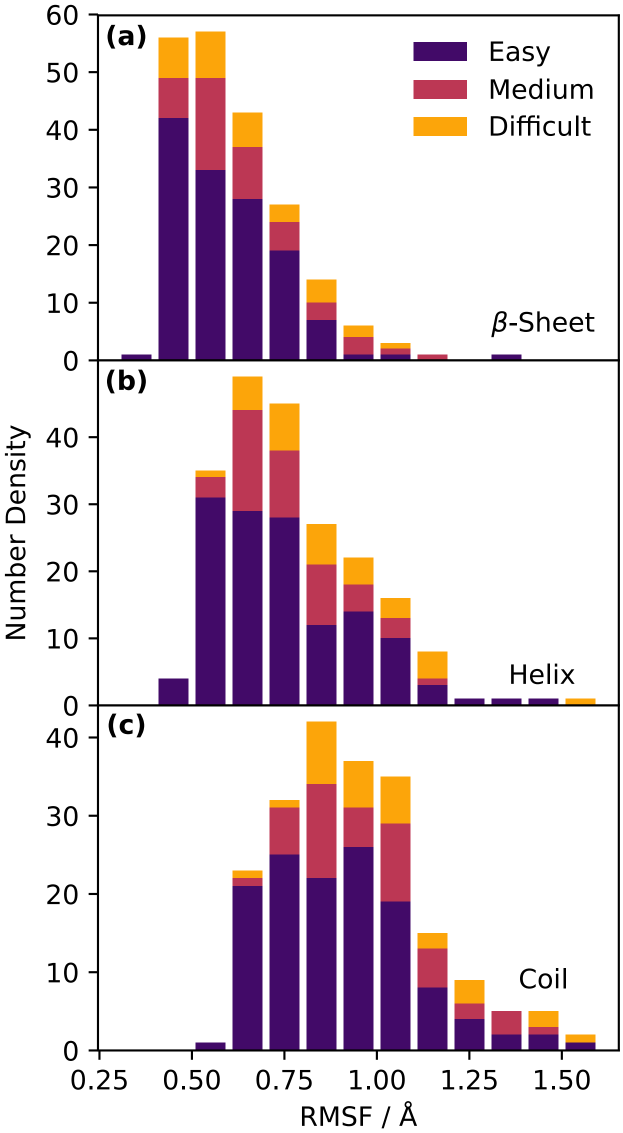

Figure 2.3.5: The average root mean square fluctuation (RMSF) for the structural elements of each protein simulated. From these specifically, we distinguish three different structural fragments: **(a)** β-sheets (DSSP classification: E, B), **(b)** helices (G, H, I), **(c)** coils (C, –, T). The classification for each protein owing to its inherent flexibility is also indicated by colour. Coil regions feature larger RMSF than sheets and helices. RMSF values are small (in most cases under 1 Å, our voxel size), demonstrating that motions captured by STID maps within 0.5 ns-long simulations are primarily due to side-chain mobility.

To include a safety margin, we run 600 ps long simulations to produce the STID maps, *i.e.*, a time at which all test cases had reached convergence. This simulation time is long enough to account for all side-chain motions but short enough that no larger conformational changes can occur. This is further demonstrated in Figure 2.3.6, which illustrates the conformational changes observed through the course of a simulation *via* the average of the all-atom pairwise root-mean-square-deviation (RMSD) of each sampled conformation.

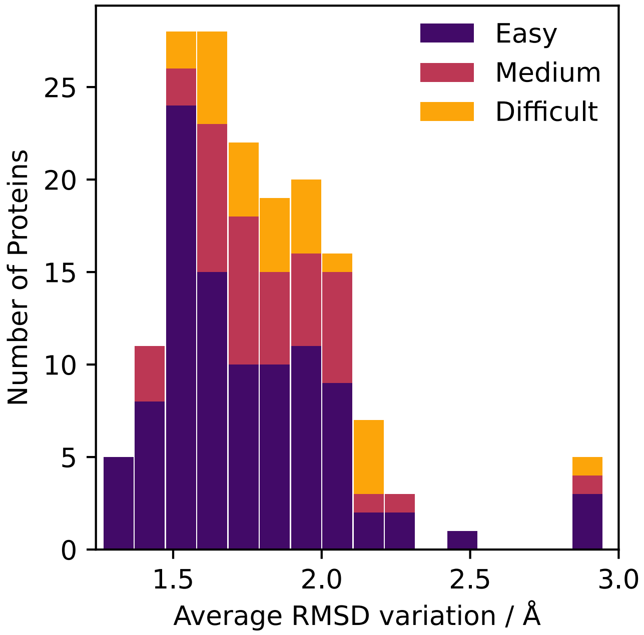

Figure 2.3.6: Distribution of individual protein RMSD variation obtained from the average of a matrix reporting on the all-atom RMSD between every conformation from the 600 ps simulations. The colouring indicates the level of inherent flexibility, with the more difficult cases exhibiting more flexibility.

The outlier at 7σ from the distributions center in Figure 2.3.4, is that of the C-terminal fragment of rabbit skeletal α-tropomysosin (PDB: 2D3E), a dimer of dimers, with each dimer subunit consisting of coiled α-helices. This protein is elongated, with a c-axis along the principal axis of the structure of 36.3 nm, *versus* 1.8 nm for the b- and c-axes. Therefore, the prolonged convergence times are likely due to a small amount of backbone wobble, with small motions at the interaction site between the two dimers propagating into larger motions distal from the site. Still, Figure 2.3.5 suggests that this backbone motion must be small, with this case sitting in the outlier region of Figure 2.3.6. Other cases that fall outside of the general distribution in Figure 2.3.6, are those that generally feature a greater proportion of unstructured regions or those that are unstable long-term as monomer units, such as the p300 TAZ2 domain bound to MEF2 (PDB: 3P57). The short time-frame of the simulation, however, prevents these monomers from unfolding from their native states.

2.3.3 Benchmarking the internal self-consistency of STID maps

For the maps to be internally consistent, we should be able to find a high CCC between a highly sampled maps, and a lower sampled maps that has been interpolated to the same size. We tested this by generating highly detailed maps at four neighbourhood sizes, N_h, *i.e.* the window size multiplied by the voxel width, with each highly-detailed map having a voxel width of 1 Å. We subsequently compared them to maps with the same N_h, but wider voxels, and therefore, smaller windows. These maps feature fewer voxels sampling each local environment overall. We calculated the CCC between the reference high-sampling 1 Å map and the low-sampling ones, to assess how much information is retained when the sampling is decreased. To make these comparisons possible, low-sampling maps were oversampled to have the same number of voxels as the high-sampling reference (*i.e.* data was replicated as necessary). For example, in Figure 2.3.7(a) N_h is held at 10 Å. The point at 1 Å voxel width, therefore, includes the properties of 10 voxels. The point at 2 Å includes 5 voxels, and so on, until the point at 10 Å voxel width considers only one voxel. The difference between the individual subfigures is, therefore, N_h.

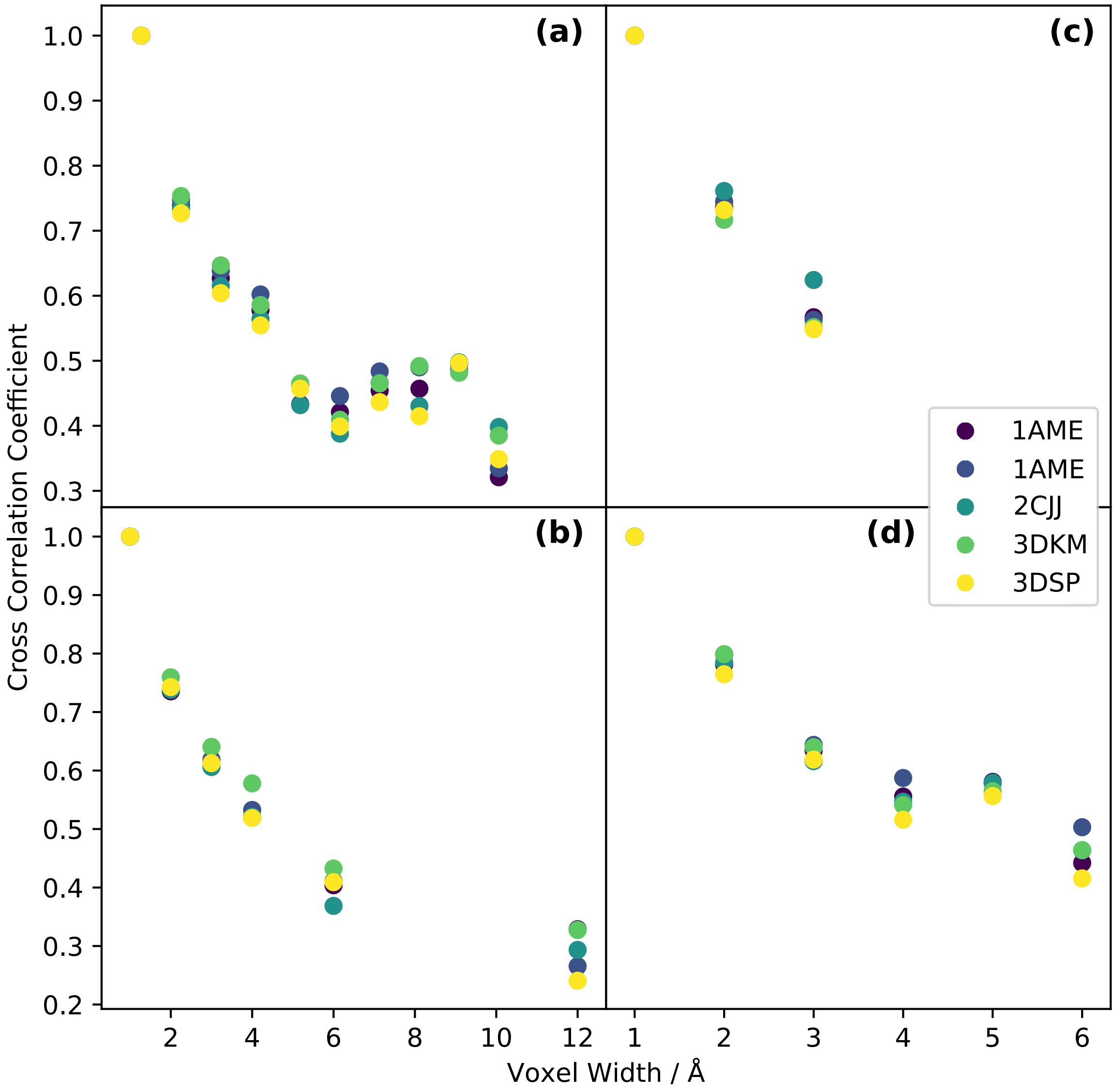

Figure 2.3.7: In each panel, we report the cross-correlation coefficient of each low-sampled map with respect to the highly-sampled 1 Å voxel width map as a function of voxel width. Five different proteins were tested, each represented by the coloured points. For each map to have the same neighbourhood size, N_h, while their voxel width is varied, the window size is adapted as necessary. Each panel represents a different N_h, according to: **(a)** $N_\mathrm{h} = 10$ Å, **(b)** $N_\mathrm{h} = 12$ Å, **(c)** $N_\mathrm{h} = 3$ Å, **(d)** $N_\mathrm{h} = 6$ Å.

The CCC between the 2 Å voxel width STID maps and the 1 Å for a variety of N_h is roughly 0.8, implying that the sampling in this region is dependable. While sampling 3 or perhaps 4 times less provides an acceptable CCC, beyond this point, the differences become too significant. An increased voxel size does affect the topology of a map. However, this is not unexpected as otherwise high-resolution electron density maps would always be interpolated from low-resolution data. The fact that the CCC remains at an acceptable level over the first few voxel widths, demonstrates that the general topology is respected by the STID maps. Thus, comparative structural data can still be obtained with different parameters, an important fact for any well-grounded method.

2.3.4 Relationship between STID maps and the structure of a protein

The global average STID value, $D_{\text{avg.}}$, provides us with a direct comparison between the different structural and dynamic characteristics of a protein. $D_{\text{avg.}}$ is the average of all voxel's individual STID values, with the exception of those containing a value of 0. The presence of both rigid and highly-charged regions within a protein contribute greater STID values to their respective voxels than a flexible or apolar residue. This is shown in Figure 2.3.8(a), where only the core regions of the protein are observed at greater isosurface cut-offs, but the more flexible regions can be seen at lower isovalues. Furthermore, we found that, while having a greater relative quantity of charged or polar residues did indeed increase the $D_{\text{avg.}}$, time-averaged dynamics and the structure had a considerably larger impact on the maps' topology.

We sought to determine whether a link exists between the characteristic $D_{\text{avg.}}$ of each protein and any of their physical quantities that are easily measurable. Using the 118 protein dataset, we observed that the ratio between SASA and molecular weight, S_{m}, is anticorrelated with $D_{\text{avg.}}$ (see Figure 2.3.8(b)). The relationship could be fitted *via* a linear least-square fit (Pearson correlation coefficient equal to -0.80):

$$D_{\text{avg.}} = 0.34\, S_{\text{m}} + 0.59. \tag{2.3.1}$$

Thus, each STID map is associated with a characteristic cut-off value, determined by the SASA and molecular weight of the protein. Important topological features of the STID maps are entirely independent of the type of protein: core secondary structure features are always visible in and around an isovalue of 0.8, and highly charged atoms become isolated from the body of the protein at more stringent cut-offs beyond 0.9. Typical structural elements are always discernible at the same isovalue, independent from the protein under study. This means that proteins with a similar mass and aspect but a different secondary structure will have a different $D_{\text{avg.}}$ value. This is because different secondary structure elements contribute to the STID voxel system in different ways, determined by their characteristic structure and dynamics. For instance, a greater number of unstructured coils will produce a more significant number of occupied voxels as the protein explores a relatively greater region of the available space, but these will have smaller associated non-zero STID values, decreasing $D_{\text{avg.}}$.

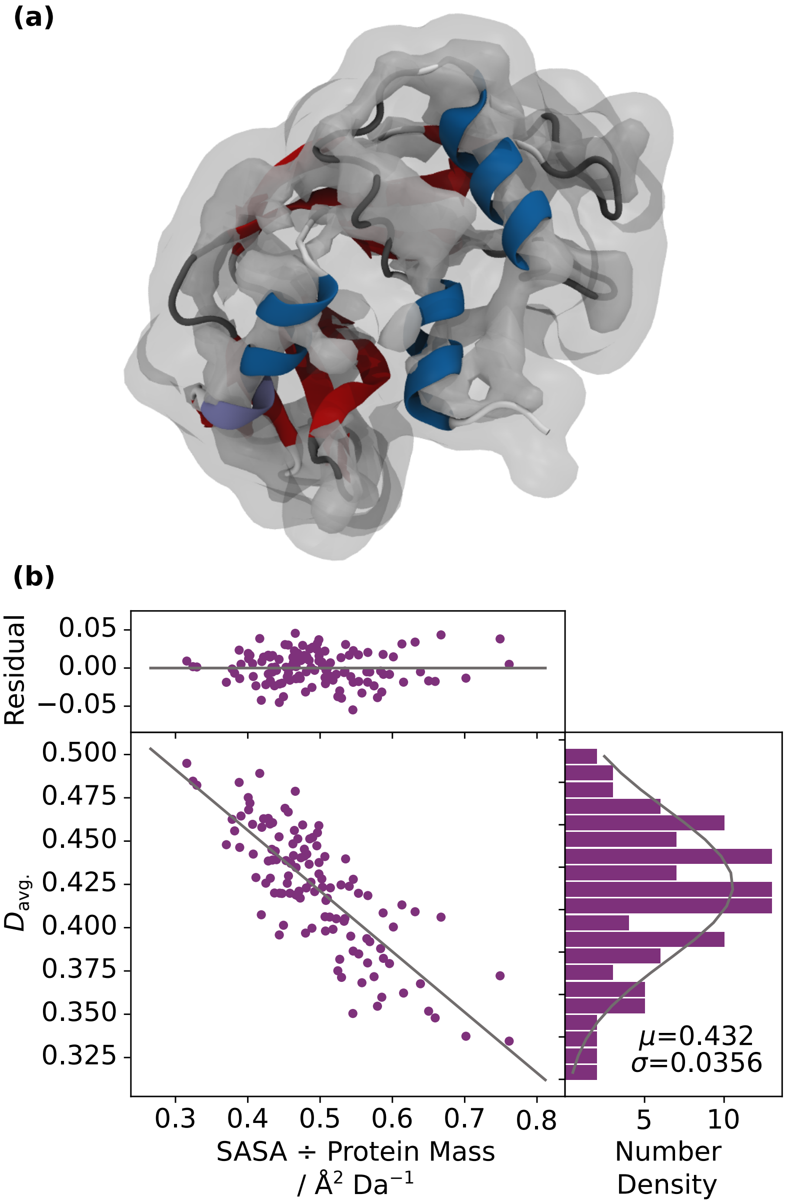

Figure 2.3.8: **(a)** Ribonuclease A (PDB: 9RSA) embedded in its associated STID map. Two isosurface selections are shown. The transparent isosurface, at an isovalue of 0.43, shows how the local side-chains contribute to the isosurface's topography. The opaque one, at 0.8, illustrates primarily the core secondary structure features and charged residues. **(b)** Bottom left: variation of average nonzero STID value *versus* the protein's solvent-accessible solvent area (SASA) divided by its molecular weight (shown in palatinate). The fitted grey line was found *via* a linear least-square fit, with a Pearson correlation coefficient of −0.80. Top: residual between the points and the fitted line. Bottom right: representation of the points as the STID average against number density in palatinate, with a fitted Gaussian shown in grey ($\mu = 0.432$, $\sigma = 0.0356$).

This direct link between the structural characteristics of a protein and the shape of the associated volumetric isosurface makes STID maps an appropriate way of representing how a protein will be perceived by its immediate surroundings, such as a binding partner in a docking scenario. Indeed, the direct relationship between $D_{\text{avg.}}$ and SASA indicates that an isosurface extracted from a map at a protein's $D_{\text{avg.}}$ is indicative of that protein's excluded volume. We can identify an ideal general cut-off value in our STID metric (an ideal isovalue) to transform all of these volumetric maps into three-dimensional shapes by taking the peak of the fitted Gaussian in Figure

2.3.8(b), which mirrors the distribution observed in Figure 2.3.1(a). This ideal isovalue of 0.43 is a property that naturally falls from the simulation in solution, specifically from the relationship between the average STID value and the SASA. In previous electron density modelling and 3D reconstruction software yielding volumetric representations, the choices of isovalue cut-off to display isosurfaces have been arbitrary.[6] Their choice is often chosen based on what is deemed by the authors to be most appropriate for the work, with no apparent link between a defining characteristic of a protein and the isosurface shown. In contrast, we have demonstrated that our STID map-based representations can be directly related to a physical and easily measurable quantity. In the remainder of this Chapter, we will discuss how our STID representations can be further utilised in a range of additional physical and biophysical scenarios.

2.3.5 Methods

2.3.5.1 Molecular Dynamics

All simulations are run on the GROMACS[17] MD engine, with Amber ff14SB force field.[18] Systems are prepared by immersing the protein of interest in a TIP3P water box, neutralised with Na^+ or Cl^- counterions. The system is then energy-minimised using a steepest descent algorithm, with a tolerance threshold set to 200 kJ mol^{-1} nm^{-1}. The initial step size is set to 1 pm, the maximum number of allowed steps to 5 $\times\ 10^6$. The cut-offs for both Coulombic and van der Waals interactions are set to 1.2 nm.

The protein is then equilibrated for 500 ps within a canonical ensemble, with T set to 310.15 K with 2 fs step size and the constraint algorithm LINCS applied to the bonds.[19] A particle mesh Ewald summation is used to treat long-range interactions and a velocity-rescale temperature coupling method applied separately to protein and nonprotein atoms; the coupling constant is set to 0.1 ps. Velocities are randomly assigned from a Boltzmann distribution of velocities at T.

Finally, production occurs over a 600 ps timescale, for reasons discussed in Section 2.3.2, in an isothermal-isobaric ensemble. T is set as mentioned above; the pressure is set to 1 bar. Berendsen temperature and pressure coupling methods are used, again keeping the protein and nonprotein groups separate. The temperature coupling is as mentioned above, with the pressure coupling constant set to 10 ps. The compressibility for both is set to 4.5×10^{-5} bar^{-1}. Atomic coordinates are saved every 5 ps.

2.3.5.2 Dataset setup & SASA parameters

In total, 118 non-redundant proteins (maximum 30% homology) were extracted from the PDB-REDO databank,[20] acting as a diverse subset of the CAPRI protein docking benchmark.[13] All structures were water-soluble proteins featuring solely standard amino acids, none required applying a biomatrix, and all were composed of more than 30 amino acids.

The SASA of each structure in the benchmark set was calculated using the Shrake-Rupley algorithm,[21] with the solvent probe radius set to 1.4 Å to represent that of water. For each protein, we report the average SASA over a 600 ps production cycle (one structure every 5 ps, excluding the first 50 ps). Molecular weights were calculated, accounting for all atoms present in the atomic structures.

2.4 Measuring microscopic permittivity

2.4.1 Introduction

Knowledge of a material's dielectric properties is important across a wide-range of fields, including the development of battery technologies,[22] new industrial processes,[23] cosmetics[24] and our understanding of dark matter.[25] These dielectric properties are characterised through the local dielectric permittivity or dielectric constant, ϵ_{r}. In the life sciences, ϵ_{r} is a key quantity in the interactome, from the interaction of amino acids that give rise to the quaternary structure of a protein, to the many intercellular processes such as neuron signalling. Variability in the relative permittivity at a molecular level is, therefore, a fundamental aspect of life, but the complex dynamics and presence of clustered charged particles results in a constant state of flux. Consequently, ϵ_{r} is difficult to calculate and predict.

It is beneficial to reconcile the dielectric properties of these non-trivial systems at a molecular level with the processes that drive biological behaviour. Measuring the dielectric permittivity at this scale is challenging, with current techniques all possessing their limitations. Since proteins are considered polarisable materials, common experimental means of characterising their dielectric properties include the use of dielectric spectroscopy,[26] impedance spectroscopy,[27] and dielectrophoresis.[28] These yield average quantities representative of the protein in bulk solution. Microscopic measurement attempts have been made on DNA using dielectric sensitive fluorescence dyes.[29,30] These provided little information on the polarisability and consequently tended to overestimate the dielectric constant.[31]

Theoretical methods can be a little more versatile. Following from the theory laid out by Kirkwood and Fröhlich,[8,9] Pitera *et al.*[11] used MD to calculate the dielectric properties of globular proteins in the absence of an applied field by exploiting the instantaneous dipole fluctuations. This approach gave numbers close to experimental values, though it neglected the microenvironment. Li *et al.*[32] were able to obtain a much higher resolution by applying their Gaussian smoothing function based on the inverse of a model for the electron density, in an approach not unlike our own for the STID maps, except with a quantum focus. However, this technique was parameterised on a set of fixed atomic structures assumed to be at 0 K, when in reality the dielectric is likely to be blurred over a localised region due to local dynamics. It also ignored the dielectric response due to local charge introduction/removal, *i.e.* the coordination between a titratable group and a counterion in a solution of appropriate pH. Thus, it is unknown how transferable it is to a realistic system. Ideally, we require a method that is easy to use, is widely transferable and affords a high resolution.

By leveraging on MD, we can address each of these key criteria. MD usage is widespread and can often run on desktop machines. The forcefields are usually very transferable, as the most adept get tested under a wide range of thermodynamic conditions at body temperatures and pressures. By its very nature, MD considers the dynamic relationship between ions and water in solution with the titratable groups of a protein. Finally, all-atom forcefields such as OPLS or AMBER[33,34] give an atomic level of resolution. In Section 2.2 we designed a representation of a protein that accounts for its dynamics, electrostatics and shape. Rather than measure the dipole fluctuations across the whole protein, we discretised the space further into a series of voxels, before convolving a Gaussian with a pseudo-atom that occupied the individual voxels. By truncating this method at the point where local permittivities were calculated (Equation 2.2.3) we can test whether STID maps are able to coarsely predict the dielectric constant, thereby providing new insight into the variation of ϵ_r that can occur within a complex, dynamic system, particularly at an interface.

As a proof of concept, we will attempt to replicate experimental permittivities for a dielectrically diverse set of solvents: ethanol, DMSO, THF, NMP, acetone and benzene, to ensure that the method is both accurate and applicable. We will first describe the theory underpinning the method, before determining the best parameters to predict the permittivities. Finally, we will discuss the results of the approach and compare with other techniques designed for similar purposes and consider the overall applicability to proteins. It is worth emphasising, however, that surpassing the accuracy of current methods designed for the purpose of predicting either protein or solvent dielectrics would be surprising, as the intended objective of the STID map is its application in protein-protein docking. Still, being able to predict physical trends using the foundational theory of the STID map would help legitimise its physical grounding, in contrast to other shape-based methods used in protein docking.

2.4.2 Measuring the dielectric constant – theory

Dielectric permittivity is a crucial physical property which characterises the electric polarisation response of a material under the influence of an external electric field. It is defined as a proportionality constant, ϵ, given by the ratio between the applied electric field, $\overrightarrow{E}$, and the corresponding electric displacement, $\overrightarrow{D}$:

$$\overrightarrow{D} = \epsilon \overrightarrow{E}, \tag{2.4.1}$$

When exposed to a field, electrical charges of the opposite sign will attempt to separate, with positive charges moving in the direction of the field and *vice versa.* At low frequencies, $10^3 - 10^9$ Hz, molecules with a permanent dipole moment will rotate and become aligned parallel to the field, a property known as dipolar polarisation. Between 10^9 and 10^{12} Hz, molecular bonds will stretch as ions get pulled apart, and beyond this, the electron cloud and nuclei begin to separate. At very high frequencies ($> 10^{15}$ Hz), the material cannot exhibit any dielectric properties due to the formation of a plasma. In bioscience, it is the dipolar polarisation that is of interest. In this frequency range, polarisation can also occur with molecules with an instantaneous dipole moment, though the extent depends on the material and the frequency of the field.

The electric polarisation, $\overrightarrow{P}$, represents the extent of separation and thus a material's response to an electric field. The following relates the field, displacement and polarisation:

$$\overrightarrow{D} = \epsilon_0 \overrightarrow{E} + \overrightarrow{P}, \tag{2.4.2}$$

where ϵ_0 is the permittivity of free space ($\epsilon_0 = 8.85 \times 10^{-12}$ F m^{-1}) which defines the relationship if the material is non-polarisable (*i.e.* a vacuum). The dielectric permittivity and electric displacement, therefore, define the extent a material becomes polarised while under the influence of an electric field. The dielectric constant of a material, ϵ_r, expresses the ratio between the permittivity in a vacuum and the measured permittivity:

$$\epsilon_\mathrm{r} = \frac{\epsilon}{\epsilon_0}. \tag{2.4.3}$$

ϵ_r is always ≥ 1 and is the quantity we endeavour to find. In the same fashion as in Section 2.2, we take the conformational space of an MD simulation as a starting point, with the space discretised into a series of voxels. The internal charge distribution of each voxel is again the explicit atoms from the MD simulation. The charges from neighbouring voxels produce the net field required for a dipolar response, and consequently a dipole associated with the voxel centre. Following Equations 2.2.1 – 2.2.3, we can use the fluctuations of each dipole moment, $\overrightarrow{M}_\mathrm{v}$, observed over the simulation to extract ϵ_v, our ϵ_r affiliated with each voxel. This approach allows us to explore the variation in the permittivity of a system on an Ångström scale. Whether we can physically measure the permittivity on an Ångström scale is worth considering, as typically the electric permittivity is treated as an averaged macroscopic property of a material. Recent AFM work has shown that it is possible to measure dielectric variation on a nanometre scale for a series of crystalline materials.[35] In addition, recent work looking into the dielectric constant of water in a confined space was able to measure the permittivity across a 3 Å film of water, where they found a value of $\epsilon_\mathrm{r} = 2.1$, owing to the almost complete suppression of the dipole rotational response in the interfacial layer.[36] Therefore, while the physicality of using a method not designed for high-resolution measurements of permittivity, for that purpose, is debatable, there is a scientific basis to measure what is typically considered a macroscopic quantity on a microscopic scale.

2.4.3 Parameter determination

We applied this theory to an MD simulation for each solvent; see the Methods section below for details on the simulation setups. When measuring ϵ_{r} for each solvent, we must also ensure we account for any discrepancies. Notably, four factors can impact our permittivity:

1. The timescales over which we measure the dipole fluctuations.
2. The external dielectric, ϵ_{Ex}.
3. The voxel volume, V_{v}.
4. The chosen forcefield.

While both a timescale and V_{v} were effectively determined in Section 2.3, it is still necessary to conduct a parallel benchmark here, both because we are attempting to represent a different physical quantity, and due to the absence of the neighbourhood factor discussed in 2.3.1. This factor is not applicable here as we are interested in a single voxel's dielectric behaviour in isolation, albeit influenced by its surroundings.

We investigated the impact of the forcefield by performing benchmarks with two forcefields for ethanol, OPLS-AA from 2006[37] and the LigParGen OPLS-AA webserver from 2017.[38–40] For the following set of benchmarks, the mean of ϵ_{r} from all the voxels is used, as we must first ensure we can accurately represent the bulk before examining a higher resolution.

2.4.3.1 Timescales

Prior to checking the other parameters, we must ensure full convergence in our simulation. Figure 2.4.1 shows how successively larger blocks of the MD simulation can be used to generate ϵ_{r}. In Figure 2.4.1 (right) we move in incremental steps of 500 ps, first using 0–500 ps, then 0–1000 ps and so on until the full 3 ns post-equilibration simulation to calculate ϵ_{r}. For this, we used a voxel width of 1 nm and an external permittivity of 60. The convergence behaviour should be identical regardless of these parameters. ϵ_{r} is very stable, with no significant deviation across any of the time segments. We, therefore, looked at smaller time frames using 2 ps increasing segments from 0–2 ps up to 0–30 ps (Figure 2.4.1 left). We see convergence for all solvents within

30 ps, thus, we can use a minimum timeframe of 30 ps to calculate ϵ_{r}. However, in order to be robust, we used a timeframe of 500 ps for the following set of results.

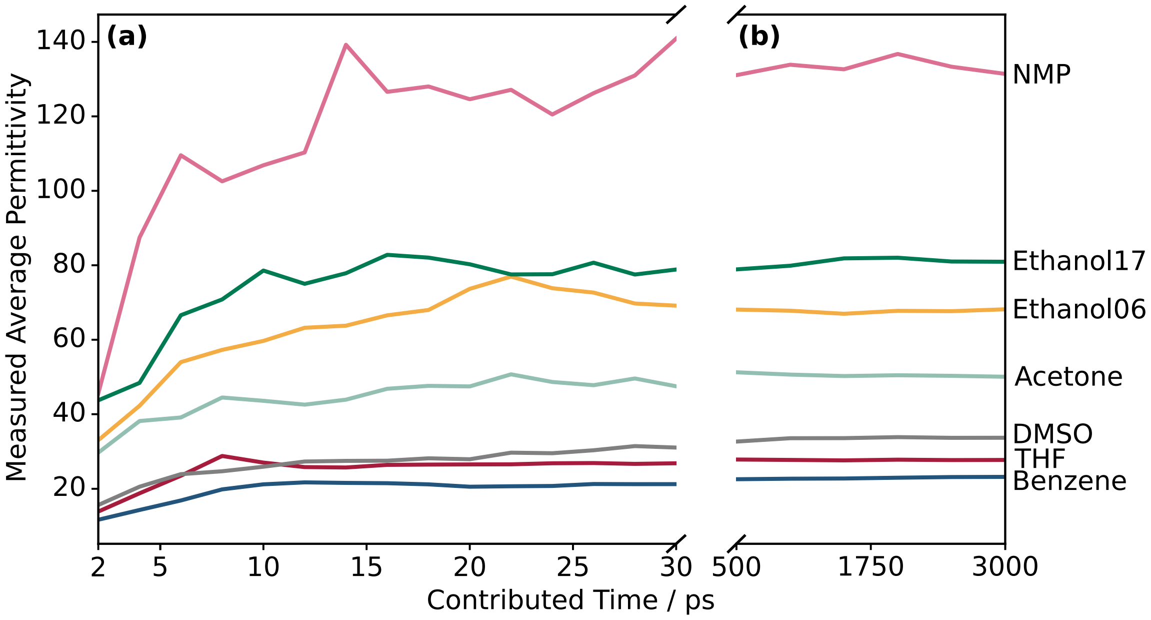

Figure 2.4.1: Measured average permittivity for each indicated solvents with increasing segments of time used to measure dipole fluctuations. The dielectric converges rapidly and is stable over long timescales. **(a)** 15 individual measurements made using 2, 4, 6 ps and so forth up to 30 ps. **(b)** Permittivity over a longer timescale, from 500 ps, 1000 ps, 1500 ps up to the full 3000 ps of the simulation.

2.4.3.2 External Dielectric and Voxel Volume

Since both the external dielectric constant and the voxel volume feature in Equation 2.2.3, it is necessary to perform a parameter sweep and test both of these in conjunction. Our choice of external dielectric should match the known value for the homogenous solvent. However, ϵ_{Ex} may not be known when exploring more complex systems, and certainly, it is not continuous at an interface. Therefore, we tested a range of values to measure its influence on the dielectric to examine whether it is possible to use an approximation or indeed fixed value for all. Specifically, we measured ϵ_{Ex} between 10 to 90.

To pair with these values of ϵ_{Ex}, we swept through a voxel width of 3 Å to 18 Å in steps of 1 Å. Since each voxel is cubic, the cube of this width gives the volume. Below this threshold, and we develop internal artefacts that arise due to pockets of vacuum, similar to the justification for our resolution pick in Section 2.3.1. Above this range, and it is not possible to exclude edge effects from the box's boundary. The measured permittivity for each combination of the two, associated with each solvent, is shown in Figure 2.4.2.

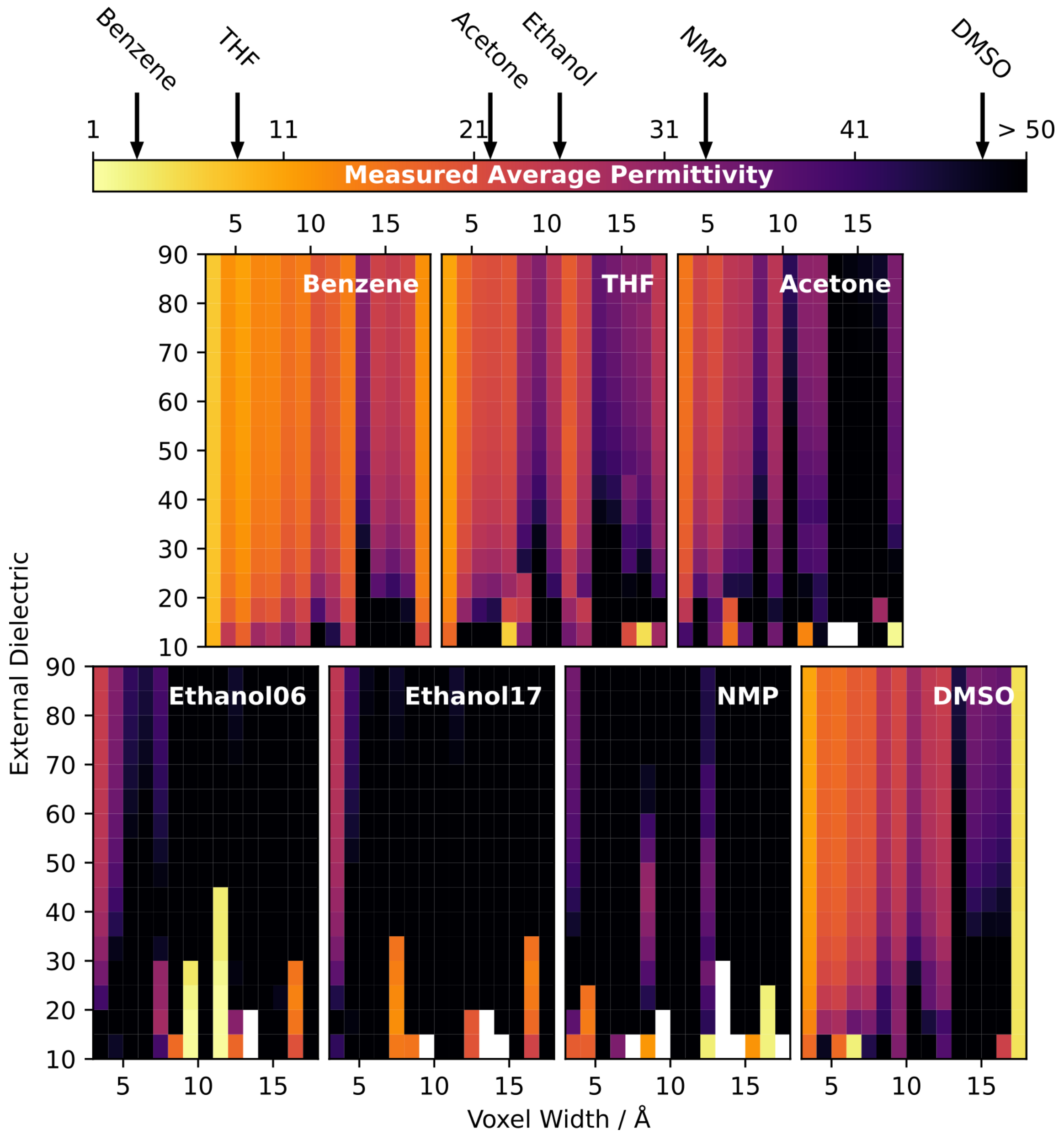

Figure 2.4.2: Variation in the measured permittivity dependent both on the voxel width and the external permittivity, ϵ_{Ex} for the seven solvents. The arrows above the colour bar indicate the experimental values.

Looking at Figure 2.4.2 it is clear that, except for DMSO, all of the solvents exhibit very large measured permittivities relative to the experimental value beyond a voxel width of 10 Å. For both ethanols and NMP, measured ϵ_{r} increases sharply beyond 4 Å. The influence of ϵ_{Ex} is more subtle, with only some minor variation along a fixed voxel width. We can therefore determine more confidently the values of ϵ_{Ex} and voxel width parameters by extracting two slices of Figure 2.4.2 at a voxel width of 3 Å and 4 Å, shown in Figure 2.4.3.

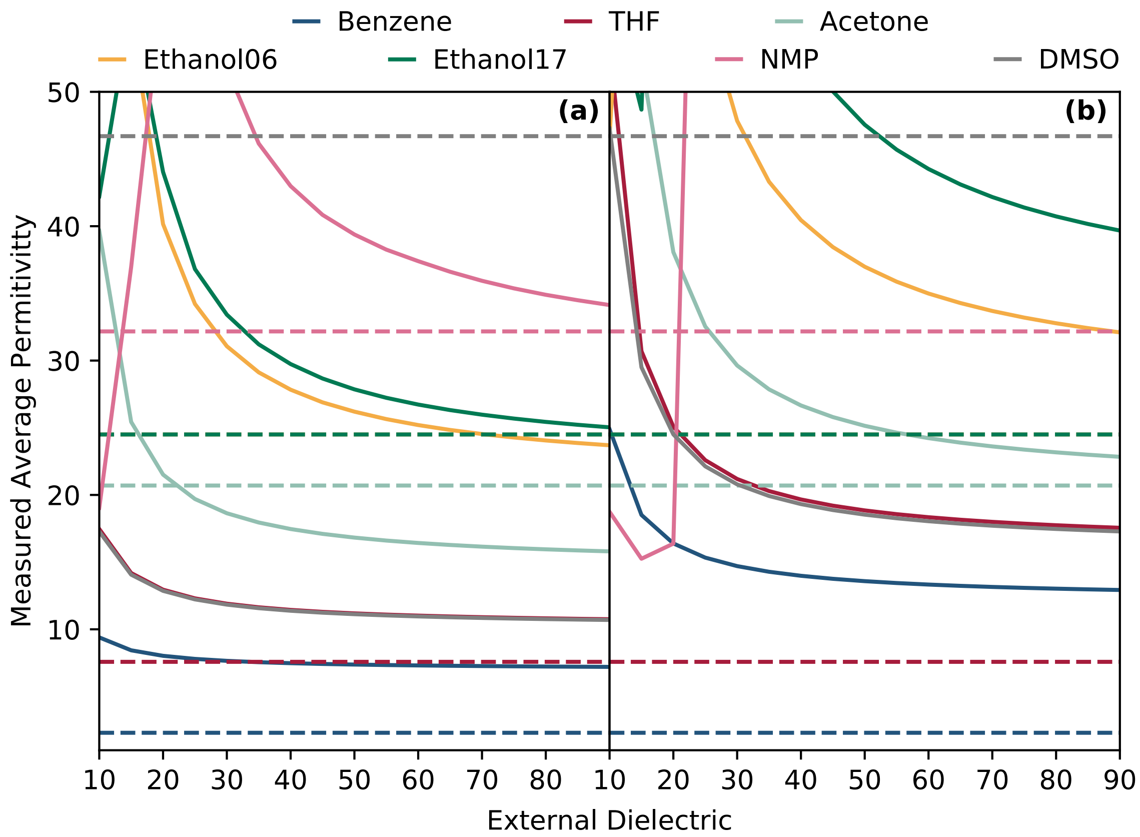

Figure 2.4.3: Variation of measured permittivity with external dielectric for the 7 solvents taken at a voxel width of: **(a)** 3 Å, and **(b)** 4 Å. The dotted lines indicate the experimental values. The green dotted line relates to both Ethanol06 (yellow) and Ethanol17 (green).

For the first few values of ϵ_{Ex}, the measured permittivity is very high, often over 100. This is particularly true for for the solvents with the larger dielectric values, except for DMSO. After an ϵ_{Ex} of 40, the measured permittivities converge to some fixed quantity that is different between the two plots. In none of the trends in Figure 2.4.3 do we see convergence to the experimental value. However, a voxel width of 3 Å (Figure 2.4.3(a)), does give values close to the experimental data in the majority of cases and is certainly an improvement over Figure 2.4.3(b).

Since the permittivity does not quite reach convergence in Figure 2.4.3, we can extend the number of measured points for a voxel width of 3 Å (Figure 2.4.4). At $\epsilon_{\text{Ex}} > 140$, ϵ_{r} for solvents such as NMP and ethanol are still decreasing, however it almost negligible. Thus, we conclude that convergence for all solvents to some fixed value occurs at $\epsilon_{\text{Ex}} \approx 140$. This quantity is unphysical, very few solvents are this polarisable, and it is not possible to increase a measured permittivity from dipolar polarisation, only decrease with electric field frequency. It might, therefore, be pertinent to consider ϵ_{Ex} more as some empirical mathematical constant than a physical quantity.

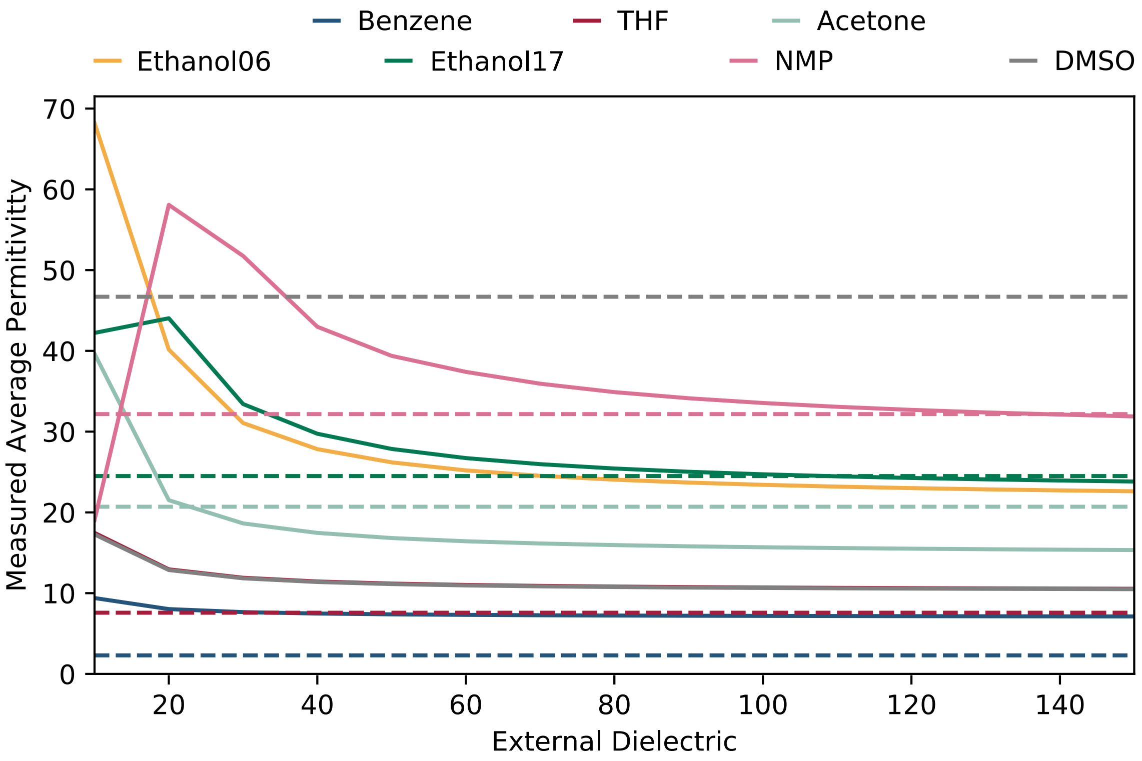

Figure 2.4.4: Variation in measured permittivity for the 7 solvents with ϵ_{Ex} over an extended range, using a voxel width of 3 Å. The dotted lines indicate the experimental values. The green dotted line relates to both Ethanol06 (yellow) and Ethanol17 (green).

Therefore, we have determined the parameters required in building our systems and the constants in Equation 2.2.3 necessary to return the most consistent and accurate results. A production timescale of 600 ps, a voxel width of 3 Å (giving a V_{v} of 27 $\mathring{\mathrm{A}}^3$), and a ϵ_{Ex} of 140. We can now consider the impact of the two forcefields used for ethanol, as well as compare more directly with the experimental data and other theoretical methods.

2.4.4 Results

Table 2.4.1 provides a comparison between our theoretical bulk mean values and the experimental data.

Table 2.4.1: Comparison of experimental and theoretical permittivity results. The standard error for the results is given in brackets. The standard deviations of the theoretical results are also provided.

Solvent	Theoretical	Standard Deviation	Experimental
Benzene	6.846(9)	1.6	2.3
THF	9.70(2)	3.6	7.58
Acetone	14.95(2)	3.7	20.7
Ethanol06	21.16(5)	8.6	24.5
Ethanol17	23.27(5)	9.0	24.5
NMP	31.9(1)	18.3	32.17
DMSO	10.29(2)	3.3	46.7

From the table, we can see that nearly all the solvents follow the expected trend and are close to the experimental data, although not all of them within the margin of error. The clear outlier is DMSO. However, OPLS-AA struggles to replicate DMSO behaviour,[41] and indeed the model comes with a cautionary note in GROMACS due to the model's proven inaccuracy. Following the results of Sweere *et al.*,[41] and ignoring the results of DMSO, we have therefore correctly predicted the relationship between increased polarisability leading to higher dielectric constants.

The standard deviation indicates that there is a wide variety of ϵ_r values found in the individual voxels. This lack of consistency implies that it will be challenging to explore the dielectric variation at a higher resolution. Thus, the method only allows us to report on the dielectric of the bulk.

2.4.5 Discussion

Work by Caleman *et al.* performed a forcefield benchmark of OPLS-AA and GAFF[42] using a different approach that relies on MD to calculate the dielectric constant on a macroscopic scale. They found that both forcefields struggled with replicating dielectric values, though GAFF performed marginally better. They note that the viscosity of the solvent will affect the permittivity measurements. Since polarisability is a contributing factor both to the viscosity of a fluid and its dielectric constant, the two are intrinsically linked. A more viscous fluid is unable to display the same degree of time-dependent fluctuations and response to the electric field as a less viscous one. Indeed, DMSO has the highest viscosity of the solvents, so is less able to react to an electric field. This issue is further exacerbated by the fact that it was simulated close to its melting point (292 K). Coupling this information with the poor performance of OPLS-AA with DMSO shown by Sweere *et al.*,[41] we can therefore omit this result and conclude that our approach accurately reflects the trend of increasing dielectric with polarisability.

Caleman *et al.* also simulated ethanol using a slightly updated OPLS-AA forcefield developed for their work and found a permittivity of 21.7. This value is a slight improvement over the 2006 OPLS-AA forcefield, with our 2017 method giving an improvement over this measurement again. Therefore the choice of forcefield is essential, and, unsurprisingly, utilising the more modern versions offers superior results. Of particular note are the polarisable forcefields based on Drude oscillator models.[43] While more specialised, these forcefields have shown to better replicate dielectric constants, particularly for highly polarisable molecules.

Deviation from experimental results and the high variance is therefore predominantly due to the low mobility of the molecules in response to the dynamic environment, a lack of polarisability and other inaccuracies arising from the forcefield. Nevertheless, our measurements are still remarkably close given that they are taken from a nanometre slice, rather than the whole of the system.

Comparing with other computational methods, Jouyban *et al.* published a method that calculated the dielectric constant of a solution based on solvent composition and temperature.[44] Their approach builds on Redlich-Kister theory; a means to calculate the physical and thermochemical properties of binary liquid mixtures based on ionic activity. They applied their method to twelve binary and three ternary mixtures. On average, they found their results deviated from the experimental results by 0.52% for the binary mixtures at room temperature, 1.29% at different temperatures and 1.61% for the ternary solvent mixtures. In contrast, our seven unary mixtures gave a deviation of 47.8%. Even discounting benzene and DMSO, we achieve 15.0%, which is significantly larger than the deviation for the more complex binary mixtures.

Considering proteins, the work by Li *et al.* on creating a model for the dielectric through a protein,[32] was able to achieve dielectric values and consequently pK$_a$s of individual amino acids close to those predicted experimentally. We initially argued that it was unknown how transferable this method is, particularly to different environmental conditions. Despite this, it offers a significant accuracy improvement over our method. Furthermore, while Li *et al.*'s approach does not accurately represent titratable groups, our method also mirrors this issue as STID maps are built solely from a protein's constituent atoms. Thus, for dielectric investigations into the nature of a protein, their work is preferable.

In conclusion, we have shown that our technique correctly predicts, in general, that increased polarisability of a molecule will lead to an increased dielectric. We have also shown that more recently developed forcefields are better at capturing the correct dielectric properties of a solvent. However, our method does not provide permittivity values of bulk solvents as accurately as methods specifically designed for this purpose. It is nevertheless encouraging to see that the theory used to build the STID map is well-grounded in physical theory, and indeed can predict important physical trends.

2.4.6 Methods

2.4.6.1 Molecular Dynamics

We carried out MD simulations of the solvents using the GROMACS molecular dynamics engine[17] with the 2006 modified OPLS-AA all-atom forcefield.[37] An additional simulation of ethanol was run using the LigParGen OPLS/CM1A parameter generator for organic ligands.[38–40] Each solvent box was cubic, at 3.5 nm across in each direction, with the typical density of each solvent at 300 K used to estimate the number of solvent molecules. Each system was initially energy minimised *via* steepest descent, with the maximum force tolerance set to 200 kJ mol^{-1} nm^{-1}. They were then equilibrated sequentially in an NVT and NPT ensemble for 1 ns, before a 3 ns production simulation in an NPT ensemble.

In all simulations, long-range interactions were calculated using the particle mesh Ewald method, and a cut-off of 1 nm was used for the van der Waals and Coulombic interactions. The LINCS algorithm[19] was used to restrain bonds involving hydrogen atoms. Simulations utilised a 1 fs integration time step, neighbour lists were updated every 5 steps. In both ensembles, the solvent was coupled to a heat bath set to 300.15 K with time constant $\tau_{\mathrm{t}} = 0.1$ ps. For the NPT simulations, an atmospheric pressure of 1 bar was maintained *via* isotropic pressure coupling using a compressability $k_{\mathrm{x}} = k_{\mathrm{y}} = k_{\mathrm{z}} = 4.5 \text{ x } 10^{-5}$ bar^{-1} and time constant $\tau_{\mathrm{p}} = 1.0$ ps.

2.4.6.2 Statistical analysis

When extracting the value of epsilon, we took the mean of all voxels not at the edge of the system. Even with the exclusion of these edge effects, we still encountered very extreme dielectric values influencing the mean. We, therefore, used the Pierce Criterion method to eliminate outliers.

Pierce's criterion is preferable to other similar methods such as Chauvenet's criterion, as it does not make any arbitrary assumptions on the rejection of data, and can also be used to remove more than one outlier. Pierces's criterion begins by assuming one outlier and a normal distribution. Based on the standard deviation of the data, and the number of observations in the sample set, we find the maximum allowable deviation of a data point. If the outlier sits outside this range, it is eliminated, and the process is reiterated, this time assuming two outliers. This process repeats until no outliers are removed, in which case we can be confident in the remaining dataset.

2.5 On the density of proteins

2.5.1 Introduction

In Chapter 1 we discussed the issue surrounding the currently accepted value of protein density, specifically the density of an individual protein, not a protein solution. The density of a protein is an essential biophysical quantity, used in a wide range of fields – most notably X-ray crystallography. This value is largely based on Voronoi tessellated generated volumes from crystal structures, yet it is treated as a constant in *in vivo* and *in vitro* studies.[45,46] Our novel protein representation, the Spatial and Temporal Influence Density (STID) map (see Figure 2.5.1) could provide much-needed clarity as to what the density of a protein is, and whether it is dependent on the local environment. We have already demonstrated that these shapes are representative of an excluded volume in solution (Figure 2.3.8), and we will discuss in Chapter 3 how they are effective molecular descriptors in a protein-protein docking context, whereby successful poses maximise surface complementarity. STID maps are therefore appropriate to investigate the relationship between protein density and dynamics. The volume a map occupies can be calculated based on the number of voxels enclosed by the surface defined by our ideal isovalue of 0.43 (see Section 2.3.4). Thus, given a STID map of a protein of known mass, we can derive a density. Unlike previous computational analyses that adopt solvent-corrected Voronoi tessellation methods,[47–49] our approach does not assume that every surface exposed atom has a well-defined position. In solution, this means that side-chain dynamics will reduce adsorption accessibility, resulting in imperfect packing of water around the protein, and thus leading to the exclusion of a larger volume. In a vacuum, the side-chain dynamics will naturally be reduced, leading to a more significant measured density overall.

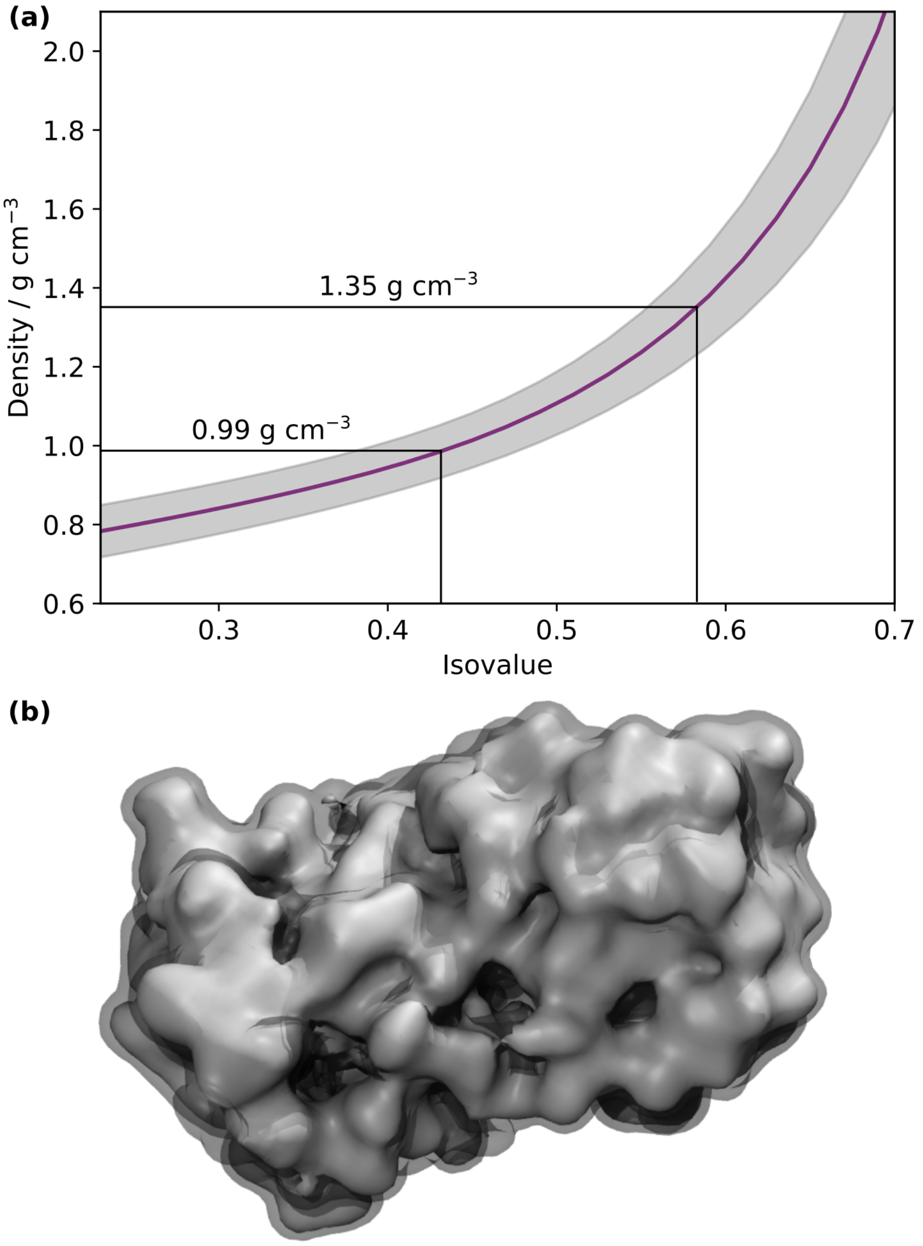

Figure 2.5.1: (a) STID map isovalue cutoff *versus* density of resulting molecular shape. The line in palatinate is the average value of the protein density for all 433 proteins used for this work, and the grey region shows the standard deviation. **(b)** Plasma protein Alpha-1-antichymotrypsin (PDB: 1AS4) embedded in its STID map. Two isosurface selections are shown. The transparent isosurface, at an isovalue of 0.43, shows how side-chain dynamics contribute to a larger excluded volume, leading to a density of 0.99 g cm^{-3}. The opaque one, at an isovalue of 0.58, negates some of the contribution made by these side-chains to give a density of 1.35 g cm^{-3}.

2.5.2 Results

We juxtaposed our method with experimental data gathered by Ashkarran *et al.*[50] obtained in solution at room temperature and atmospheric pressure,[51] by applying our STID map-based technique to calculate the density of five plasma proteins studied by them (hereon “Dataset A”) under the same conditions. These are immunoglobulin heavy constant gamma 1 (PDB: 1AQK), Alpha-1-antichymotrypsin (PDB: 1AS4), immunoglobulin kappa constant (PDB: 1D5I), Serotransferrin (PDB: 1RYO), and Immunoglobulin heavy constant gamma 3 (PDB: 5W38). We found an average density of 1.01(5) g cm^{-3}, which is in excellent agreement with their experimental value of 1.03(2) g cm^{-3}. For a greater understanding of protein density distribution, we then simulated and analysed a larger dataset of 416 proteins in solution from the established protein docking benchmark[13] (hereon "Dataset R"), as well as the 12 proteins Fischer *et al.*[7] used to identify a dependency between density and molecular weight (hereon "Dataset F") again in solution and at room temperature and atmospheric pressure. The protein density average for the total of 433 proteins is 0.99(7) g cm^{-3}. This value is substantially smaller than the typically used 1.35 g cm^{-3}, but again consistent with the recent results of Ashkarran *et al.*. To compare with experimental data from ion mobility experiments, we utilised simulation data from Mandl *et al.*[52] and Marklund *et al.*[53] of lysozyme, ubiqutin, CTF and Trp-cage (hereon "Dataset E") in vacuum and solvated by a single water shell of 6 Å, at 300 K and low temperature (200 K) conditions. We also utilised simulation data of fully solvated systems of this dataset at 200 K from Mandl *et al.*[52] and Marklund *et al.*[53] To ensure that the gas-phase simulations of Dataset E are equilibrated, we monitored how protein density evolved over a 10 ns lysozyme simulation. We calculated protein density in 2.5 ns–long windows, taken every 250 ps. Results show that the density displays a small amount of time-independent variation, and notably, no significant change in the density occurs through the simulation (see Figure 2.5.2). This result confirms that gas-phase equilibration phenomena such as side-chain collapse are not impacting our density measurements.

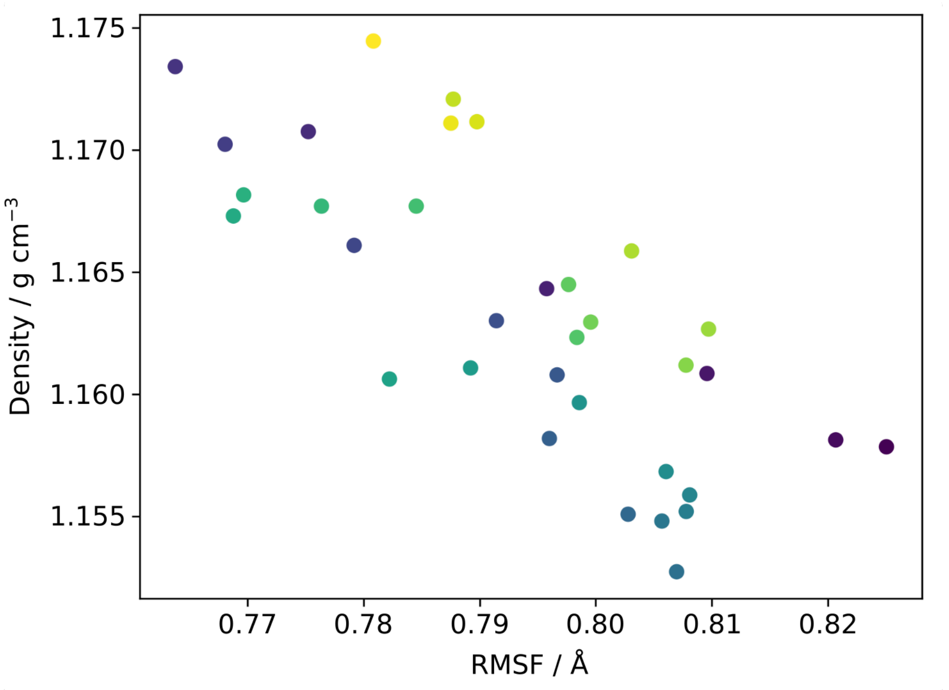

Figure 2.5.2: Density measurements from a sliding 2.5 ns block of a 10 ns simulation of lysozyme in vacuum, coloured from purple to yellow according to their progress through the simulation. Each block moves 250 ps forward. We observe no time-dependent evolution of protein density throughout the simulation.

The average density in vacuum at 300 K is 1.17(5) g cm^{-3}. A breakdown of all values from each dataset is provided in Table 2.5.1.

Table 2.5.1: Summary of the average protein densities for each dataset alongside the conditions they were simulated in. Densities are calculated through the volume encompassed by a STID isosurface at an isovalue of 0.43 for each protein. The brackets indicate the standard deviation of the measurements. Specifically: Dataset A is Ashkarran *et al.*'s five plasma proteins,[50] Dataset R are the 416 proteins from the protein docking benchmark,[13] Dataset F is Fischer *et al.*'s 12 proteins used to determine their mass dependency relationship,[7] and Dataset E is lysozyme, ubiquitin, CTF and Trp-cage at 300 K and lysozyme and ubiquitin at 200 K.[52,53]

Dataset	Ref.	Number of Proteins	Temp. / K	Conditions	Average Density / g cm^{-3}	Average RMSF / Å
A	50	5	310.15	Water	1.01(5)	1.1(3)
R	13	416	310.15	Water	0.98(7)	1.1(2)
F	7	12	310.15	Water	1.04(6)	0.9(2)
E	52, 53	4	300	Vacuum	1.17(5)	0.85(12)
E	52, 53	2	200	Vacuum	1.24(4)	0.56(6)
E	52, 53	2	200	6 Å water shell	1.36(3)	0.37(1)

To investigate whether the observed distribution of protein densities could be explained by protein dynamics, we calculated the average root mean square fluctuation

(RMSF) of each protein from the MD simulation used to generate their STID maps (see Figure 2.5.3(a)). We observed that protein dynamics and density are anti-correlated and that an exponential model could be fitted to their distribution:

$$\rho = 1.33 \exp(-0.272 \times R), \tag{2.5.1}$$

where ρ is the density in g cm^{-3} and R the average RMSF in Å. We tested the validity of this model using the Leave One Out Cross-Validation method, resulting in a training mean squared error (MSE) of 0.00196 g cm^{-3} and a test MSE of 0.00204 g cm^{-3}. The similarity between the two and their low values give confidence to our fit. Low-density outlying proteins are either highly non-globular or unstable when not in a complex. Extrapolating to the limit of a hypothetical protein with no dynamics, we find a density of 1.33(7) g cm^{-3}, which is consistent with the theoretical calculations based on protein crystal structures.[47–49]

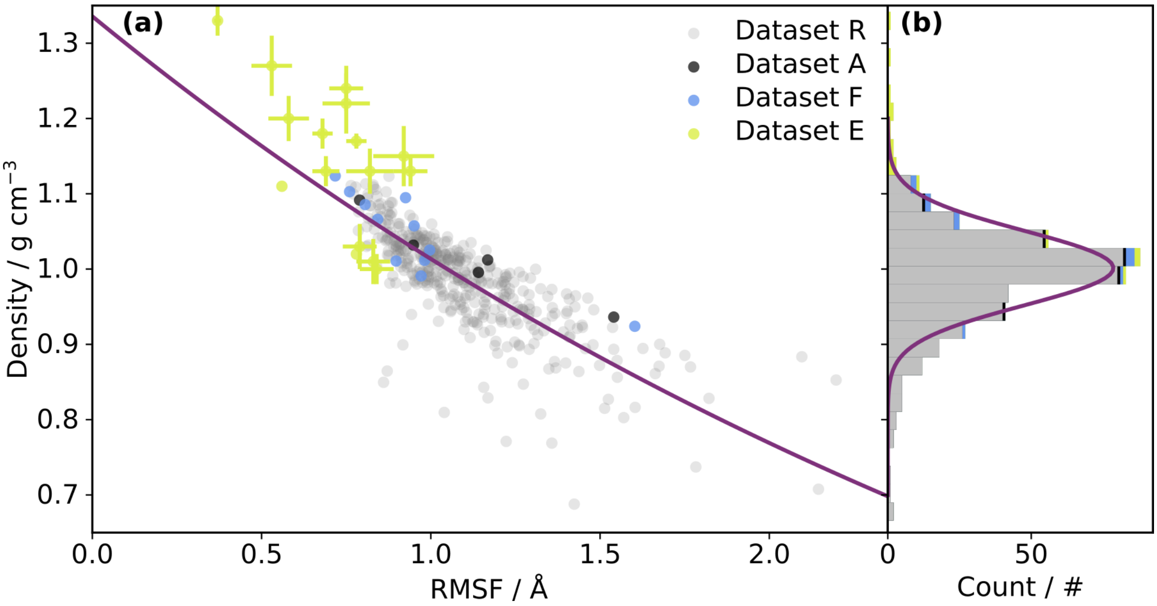

Figure 2.5.3: Relationship between protein density and dynamics. **(a)** We used our STID map representation to calculate density and average root mean square fluctuation (RMSF) for 433 different proteins and fitted an exponential model to the resulting distribution (palatinate line). The grey points are the 416 proteins from the protein docking benchmark[13] (Dataset R), the black points are five plasma proteins studied by Ashkarran *et al.*[50] (Dataset A), the blue points are 12 proteins collated by Fischer *et al.*[7] (Dataset F), and the green points (Dataset E) lysozyme, ubiquitin, CTF and Trp-cage.[52,53] The error bars on Dataset E are the standard deviations found from either 50 replicas of the simulation or independent time blocks of a 10 ns simulation. **(b)** Histogram of all protein densities fitted with a Gaussian.

We compared our results with data obtained *via* the *gmx sasa* tool[54] supplied through GROMACS,[17] a popular method based on Connolly surfaces (see Methods Section 2.5.4.1). This method requires the definition of a suitable probe size, which is uncertain in this context. For this reason, we obtained a bracket of possible density values by analysing all conformations of proteins in Dataset R using the limits of typically utilised probe sizes, 0.7 Å and 1.4 Å.[55] Consistent with our STID map-based results, we obtained average densities of 1.13(2) g cm^{-3} and 0.90(4) g cm^{-3}. These densities also featured a discernible anticorrelation with respect of RMSF, although less clear than that observed using our method (see Figure 2.5.4).

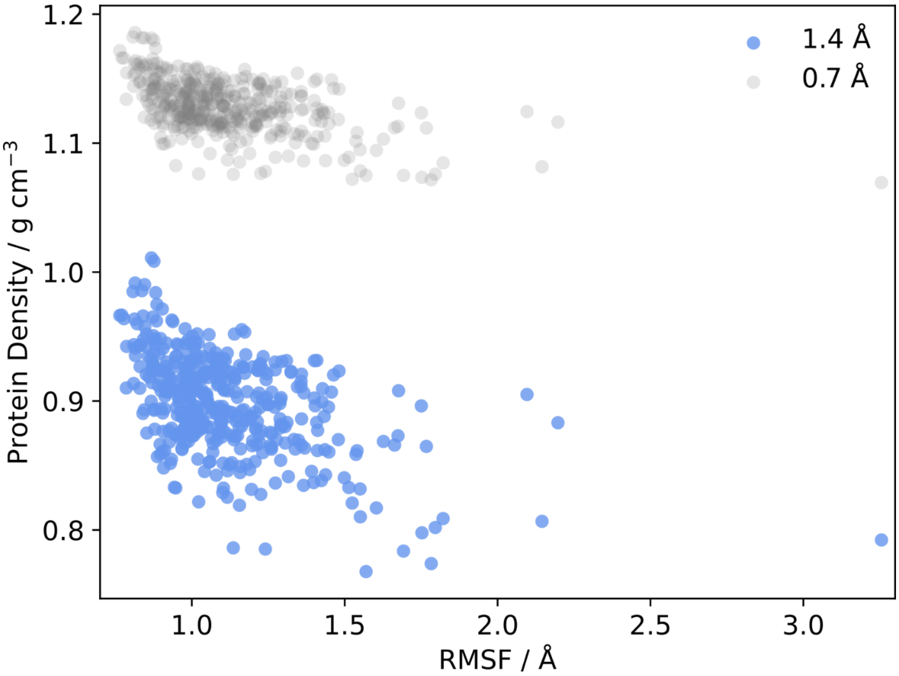

Figure 2.5.4: Protein density as measured using the GROMACS analysis tool *gmx sasa*[54] against average RMSF for Dataset R. SASAs were measured using two probe sizes (0.7 Å and 1.4 Å) representing the limits of probe sizes used for similar density calculations.[55] Their respective average protein density values bracket the density value found using our STID map-based method. An anticorrelation, albeit less clear than that obtained using our method (see Figure 2.5.3), can be discerned.

We then investigated the relationship between protein density and molecular weight previously identified by Fischer *et al.*. We first emulated, on our larger dataset of 433 proteins, calculations by Tsai *et al.* using Voronoi tessellation on single structures. To this end, we calculated densities using the flood-fill algorithm in ProteinVolume,[56] with a probe size of 1.4 Å, as a proxy for the Voronoi tessellation method used by Tsai *et al.*[48] While the ProteinVolume method itself does not apply Voronoi Tessellation, it is similar in that it does not consider the solvent-excluded volume on the surface of the protein, in contrast to the *gmx sasa* tool (see Methods Section 2.5.4.1). The densities of crystal structures from the PDB are reported in Figure 2.5.5(a) to better compare with the literature. We also calculated the densities of post-equilibrated proteins following the MD simulation. Unlike the raw crystal structures, these relaxed structures also included hydrogen atoms. On average, we observed a difference of only 1.8% between these two datasets, indicating that both equilibration and the presence of hydrogens have little impact on protein density. This ProteinVolume approach rendered a trend similar to that identified by Fischer *et al.* (Figure 2.5.5(a)). However, using densities derived *via* our STID maps (Figure 2.5.5(b)), we could no longer see any discernible trend, indicating that protein density is not a molecular weight-dependent property when dynamics are considered.

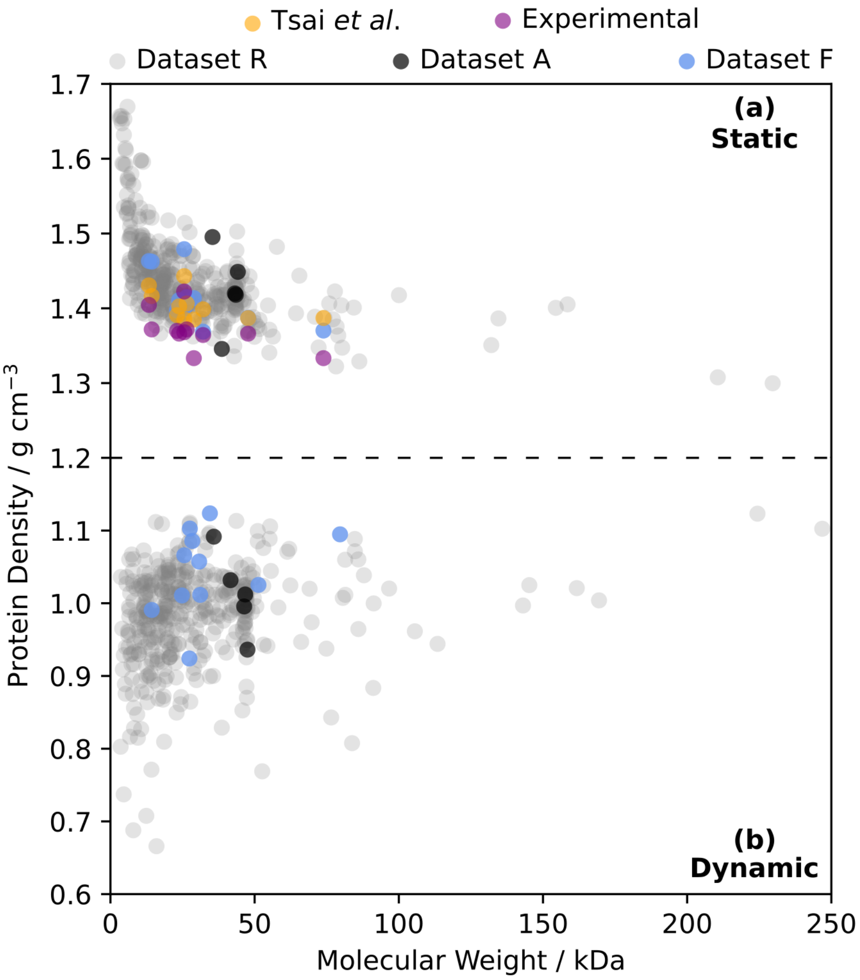

Figure 2.5.5: Molecular weight dependency of protein density. **(a)** The grey, black and blue points — proteins extracted from the protein docking benchmark,[13] Ashkarran *et al.*[50] (Dataset A) and Fischer *et al.*[7] (Dataset F) — are densities of structures downloaded from the protein data bank, that we calculated using ProteinVolume.[56] The orange and purple points show the Dataset F protein densities originally calculated by Tsai *et al.*[48] using Voronoi tessellation and experimentally determined by Squire & Himmel,[57] and Gekko & Noguchi.[58] **(b)** Dynamic protein density found using the STID map method on the 433 proteins featured in this work. The discrepancy in Molecular Weight between static and dynamic cases arises from the lack of hydrogens in the crystal structure.

Instead of molecular weight, we observed that the protein density and sphericity[59] are correlated, under the same environmental conditions (see the top of Figure 2.5.6). Dataset E contains simulations at different temperatures and pressures, hence the spread in the density at similar sphericity (see Section 2.5.4.2). Since buried amino acids typically have lower RMSF than solvent-exposed ones, proteins with higher surface-area-to-volume ratio will tend to feature larger average RMSF (see bottom of Figure 2.5.6). Consequently, an aspherical protein will tend to have a lower density than a globular protein of equal mass.

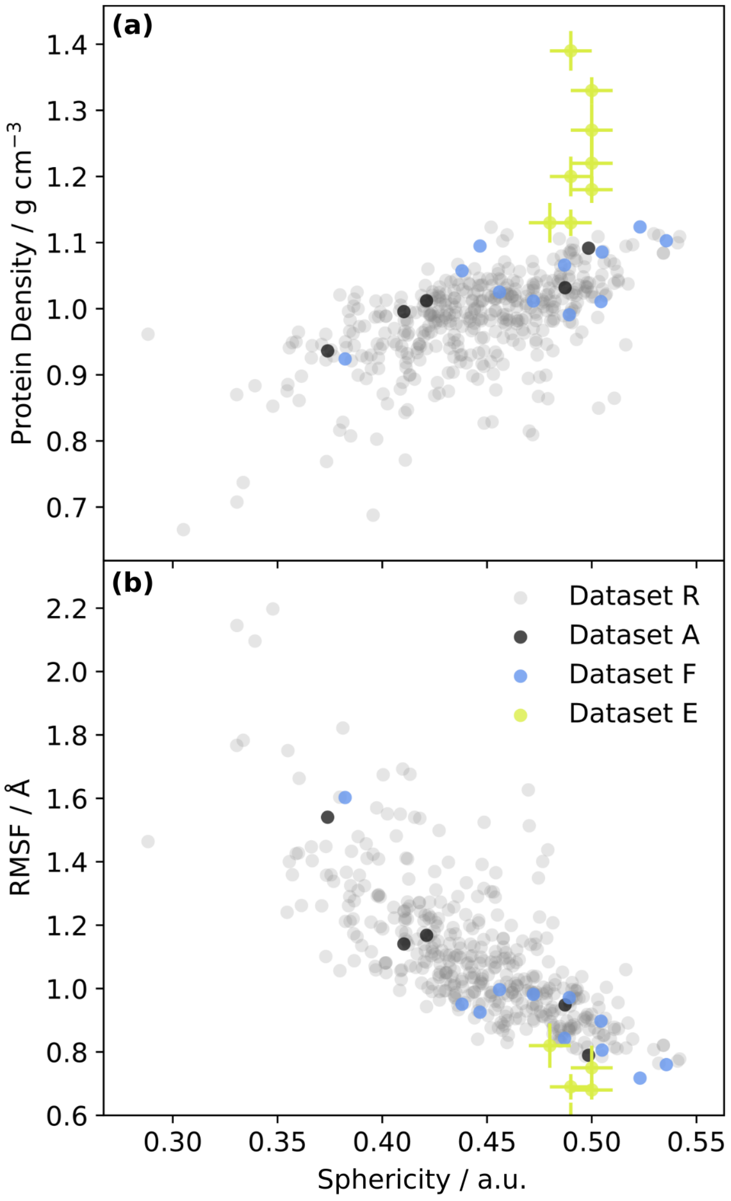

Figure 2.5.6: Relationship between protein density, dynamics and sphericity. We calculated the protein density using our STID map representation, dynamics refers to the average RMSF from the MD simulations used to produce the STID maps, and sphericity was calculated from the snapshot in the last frame of each simulation (see Methods Section 2.5.4.2). Point colours are defined as per Figure 2.5.3. As this trend is environment-dependent, Dataset E measurements do not follow the pattern of all other datasets, but are retained in the figure of completeness.

Finally, to investigate the impact of the surrounding conditions further, we also explored the effect of increasing pressure on lysozyme in solution at 300 K. The results show that an increase in pressure leads to a reduction in RMSF and consequently an increase in density (see Table 2.5.2).

Table 2.5.2: Effect of pressure on protein density. These calculations were performed on 1 ns block averages of a 10 ns simulation of lysosyme (PDB) at 300 K in water. The brackets indicate the standard error over the 10 blocks.

Pressure / atm.	Density / g cm^{-3}	RMSF / Å
10	1.03(1)	0.79(2)
1	1.01(1)	0.83(2)
0.1	1.00(1)	0.84(2)

2.5.3 Discussion

The simulations in vacuum give larger densities than those found in solution. These values are slightly larger than suggested experimentally by ion mobility results,[60] but follow our expected trend (see Figure 2.5.3). In vacuum, the proteins contract slightly, and the side-chains exhibit slower, more restricted dynamics. This contraction, therefore, reduces the measured RMSF. At 200 K, the RMSF is diminished further, both in vacuum and water, and thus the density increases. Simulations from Dataset E included a 6 Å shell of water at 200 K, where the solvent's reduced degrees of freedom decreases the competition for binding to the side-chains. Consequently, the solvent almost immobilises the side-chains, resulting in a density of 1.36(3) g cm^{-3}. This result matches the theoretical measurements performed on static structures. Correspondingly, we also found that an increase in pressure leads to a reduced RMSF and, therefore, increased density. While this was a preliminary test on a small dataset and larger pressures should be applied to confirm the trend, these results suggest the same general relationship to that observed in Figure 2.5.3. These results highlight the influence of the environment on the density of a protein, and explains why measurements performed on single-crystal structures are not appropriate as a proxy for the density of a protein in solution. The density of a protein is dependent on its environmental, and thus experimental conditions. We have shown that our density-dynamics dependency model is robust through different environments, in contrast to previous models. Since protein density is related to its excluded volume, it is typically considered a product of a proteins interaction with its environment and is thus ill-defined in the gas phase. Our model provides a novel means to define protein excluded volume, and therefore its density, in vacuum.

Using our STID map representation, we have shown that the density of a protein is no fixed value, but can be reliably predicted by including dynamic information. For the density in solution at room temperature, we find a significantly reduced average from the commonly used value of 1.35 g cm^{-3} to 0.99(7) g cm^{-3}. This value is consistent with new measurements performed by Ashkarran *et al.* in solution.[50] According to our model, protein density does not depend on molecular weight but instead on shape and dynamics, respectively quantified by sphericity and RMSF. Extrapolating our data into the regime of no internal dynamics, we find a distribution of densities in agreement with Fischer *et al.*[7] Thus, we have obtained a model based on dynamics which matches experimental data performed under different conditions. While the traditional 1.35 g cm^{-3} remains suitable for measurements performed under conditions reducing protein dynamics, such as low temperatures or crystalline phases, we suggest that the average density for proteins in solution at room temperature should be revised to 0.99(7) g cm^{-3}. A more accurate density value for a protein of interest in different environmental conditions can be calculated using Equation 2.5.1, provided the average RMSF extracted from a short MD simulation, or based on the volume occupied by the STID maps generated by our software.

2.5.4 Methods

See Section 2.3.5.1 for details on the MD setup used for this work.

2.5.4.1 *gmx sasa* calculations

We used the *gmx sasa* analytics tool[54] available through GROMACS,[17] on all proteins of Dataset R. This tool finds the SASA of a protein for each frame in a simulation and has been previously used to estimate the density of a protein.[55] For our work, we used the two probe sizes of 1.4 Å and 0.7 Å as these have been indicated to be the necessary limits.[55] The default probe size of 1.4 Å suits conventional SASA calculations in water, but the choice of 0.7 Å seeks to define a surface closer to the atomic nuclei in the protein. *gmx sasa* can also be used to calculate the volume enclosed by the surface generated which, combined with the mass, can be used to find a density. For each reported protein density, we averaged the full 600 ps to compare directly with the STID maps.

2.5.4.2 Sphericity calculation

We calculated the sphericity using ProteinVolume,[56] with a rolling ball probe of 3.0 Å. This probe size was chosen following work by Kim *et al.*,[59] who determined how best to calculate sphericity. The surface area was found using the Shrake-Rupley[21] algorithm with the same probe. The sphericity was then calculated using the following:

$$S = \frac{\pi^{1/3}(6V)^{2/3}}{A}, \tag{2.5.2}$$

where S is the sphericity, V the volume and A the surface area. S equal to 1 indicates a perfect sphere, 0 an infinite plane.

2.6 Providing insight into residue structure and dynamics

2.6.1 Introduction

As emphasised in Chapter 1, the binding of proteins, facilitated by the structure's motifs, provide the necessary complexes required for function, or indeed malfunction. Binding can occur across multiple scales: from the aggregation of transthyretin ($<$ 1 kDa) into amyloid fibrils responsible for amyloidosis,[61] to the interaction of the SARS-CoV-2 spike glycoprotein ($>$ 420 kDa) with the ACE2 receptor in lung cells, the first critical stage in viral entry, and therefore key to the virus' overall life cycle.[62]

When binding occurs with larger proteins, it is usually accompanied by a conformational change, potentially localised at the binding site, but often over a larger scale, with movement far from the interaction site. These shifts can be anything from a single residue displaying different dynamics, to large global reshuffling of tertiary structure. This allostery manifests as distinct unbound and bound states for a protein. In the SARS-CoV-2 spike glycoprotein, the spike adopts distinct pre- and post-fusion conformational states, with the pre-fusion unbound state required for ACE2 binding, and the latter post-fusion bound state needed to begin the process of virus endocytosis.[63] The specific pathway that describes rearrangement between the states is complicated and not well understood. Indeed, in many analogous proteins, there is a poor comprehension for how or which distal domains or residues can be mechanistically connected without extensive experimental work, highlighting the need for a computational technique.

Recent work by Ho and Hamelberg[64] developed a novel method to automatically identify if and how distant domains of a protein are coupled by applying a Markov-Ising model to a set of Molecular Dynamic replica simulations. In their work, they used bovine pancreatic trypsin inhibitor (BPTI) as their test structure, utilising a very large (1024 copies of 100 ns) simulation by the D. E. Shaw research group on the specialised MD supercomputer Anton.[65] Ho and Hamelberg's method identified three key domains in BPTI (see Figure 2.6.1), where each could exhibit a binary on or off state depending on their conformation. Their algorithm endeavoured to resolve the transition probabilities between a domain's two states after 100 ns based on the on/off initial states of the other two domains. This would confirm which domains, and by extension which groups of residues, were coupled.

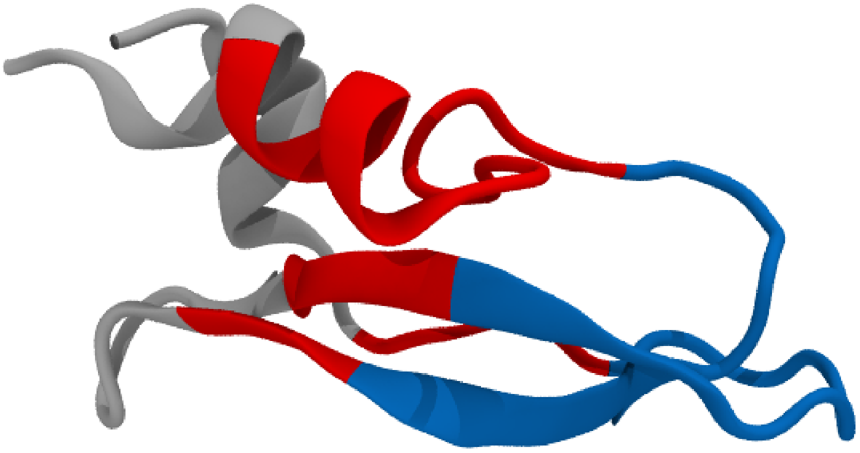

Figure 2.6.1: The three domains of BPTI as determined by Ho and Hamelberg[64] using their Markov-Ising model.

They concluded that while the silver and red domains had some dependency on each other and blue, the blue domain had no dependency on any other domain's state. We note that, while not discussed in their work, the blue domain contains the BPTI binding site across all complexes it is involved in. Thus, it is likely the blue domain's state following the formation of a complex is important for the conformation and dynamics of the neighbouring domains. BPTI, therefore, serves as an excellent minimal example for the conformational switch phenomenon exhibited by much larger proteins such as the SARS-CoV-2 spike glycoprotein. It is also a particularly good example for this work due to the number of complexes it is involved with.

Extending Ho and Hemelberg's sophisticated model to much larger proteins would likely incur a significant computational cost; due to the required simulation length times, the number of replicas needed, and the fact that the number of modes the model must accommodate for scales as $O(n^2)$ with the number of domains. In addition, while their method identified links between domains, it neglected the residue level changes. Thus, while this method provides a great deal of information regarding transition probabilities, and how domains are connected, it is desirable to have a higher resolution, quicker, and less costly method to better illustrate how binding impacts conformational state occupation and associated dynamics.

The STID map isosurfaces offer a means to characterise these changes. As shown in Section 2.5, the topography of an isosurface extracted from a STID map is sensitive to its environment. Another critical advantage of these isosurfaces, is that they report on structural information, specifically on how local dynamics can alter the adopted residue conformational states. Thus, in addition to utilising STID maps to extract information about a protein from simulations in solution *versus* vacuum, we can also investigate whether STID maps can provide insight into how the formation of a protein complex with a specific receptor might impact the dynamics of the composite proteins relative to another bound complex, or indeed a protein in isolation.

By simulating BPTI in several complexes, *i.e.* with different receptors, we can build a STID map based on the captured dynamics of BPTI in each of these bound states to elucidate which domain's dynamics are impacted by the binding. We can use the difference between two STID maps (see Section 2.6.4.2) to highlight regions of interest. Given that we are interested in a high-resolution method to assess the impact of binding on structure and dynamics, we will compare STID maps of individual residues extracted from BPTI in different complexes to explore how the orientational states of residues are affected on the timescale of side-chain dynamics. Thrombin is known to induce significant structural rearrangement in BPTI.[66] Thus, we will compare a diverse range of 14 other bound complexes and the unbound BPTI with BPTI extracted from the thrombin-BPTI complex (PDB: 1BTH).

2.6.2 Results

In total, we generated 17 individual simulations of BPTI. One of the unbound wild-type case (PDB: 9PTI), 14 of BPTI bound with various receptors, and two of BPTI bound with thrombin (PDB: 1BTH). The 1BTH simulation was repeated such that we could compare and contrast which dynamic behaviour is due to the different binding interactions with a receptor, or lack thereof in the unbound case, and which is stochastic. Table 2.6.1 gives the PDB codes, and binding partner names where appropriate, for each BPTI case. Following alignment, we compared the STID maps of each BPTI residue extracted from its bound or unbound simulation *versus* the reference 1BTH BPTI residue STID map.

Table 2.6.1: PDB codes and corresponding receptor, where appropriate, for the various BPTI cases. Two simulations for 1BTH, highlighted in bold, were generated, the first as a reference and the second to quantify the stochastic dynamic behaviour of the protein. The 9PTI case is the unbound wild-type of BPTI.

PDB Code	BPTI Binding Partner
1BTH	**Thrombin**
1BZX	Anionic Salmon Trypsin
1CBW	Bovine Chrymotrypsin
1EAW	Matripase
1FY8	Anionic Trypsin II
1TPA	Cationic Trypsin
1YKT	Trypsin II
2IJO	West Nile Virus Protease
2KAI	Porcine Kallikrein A
2R9P	Trypsin III
2RA3	Trypsin I
3U1J	Serine Protease
4DG4	Human Mesotrypsin
4WWY	Trypsin I mutant
5YVU	Dengue Virus Protease
9PTI	–

Figure 2.6.2(a) shows how the fraction of dynamics that are different relative to the 1BTH BPTI reference, F_{D}, changes across each of the other 16 BPTI cases for each residue in BPTI. A value of $F_{\mathrm{D}} = 0.0$ measured between a specific case and the 1BTH reference, indicates that a residue behaves identically to its equivalent in 1BTH BPTI. $F_{\mathrm{D}} = 1.0$ shows that there is no overlap whatsoever between the conformational states of the two residues (see Section 2.6.4.2 for how F_{D} is calculated). The secondary structure associated with each residue is indicated above the figure, with C referring to unstructured, H to α-helix, and E to β-sheet. The figure is split into sections by colour according to the domain of the residue. The 14 BPTIs bound in various complexes (*i.e.* not the unbound case or the 1BTH repeat), were averaged to give $F_{\mathrm{D,avg.}}$. Therefore, the vertical lines shown in the bottom of Figure 2.6.2 represent one standard deviation of these 14, while the midpoint is $F_{\mathrm{D,avg.}}$. Also provided, is the value of F_{D} for the unbound BPTI case ($F_{\mathrm{D,9PTI}}$), shown as a circle, while the 1BTH BPTI repeat ($F_{\mathrm{D,1BTH}}$) is shown as a cross. Both $F_{\mathrm{D,9PTI}}$, and the standard deviation of F_{D} across the 14 bound cases, allow us to determine whether any discrepency in $F_{\mathrm{D,1BTH}}$ is due to the stochastic behaviour of the simulation, or whether there is some underlying mechanism or interaction that has directly impacted the conformational

space of a specific residue.

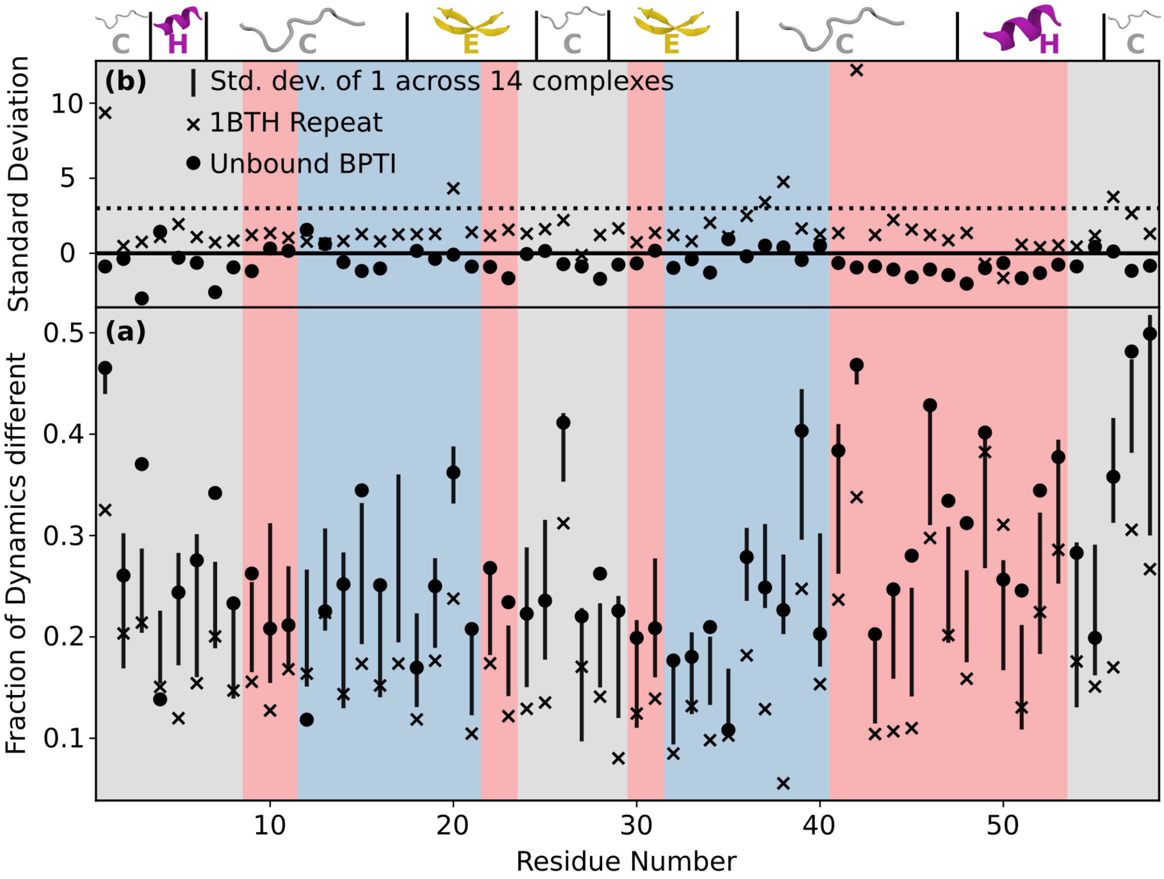

Figure 2.6.2: **(a)** The fraction of BPTI residue conformational dynamics different between any specific BPTI residue and the BPTI extracted from the 1BTH reference (F_{D}). We show the average across all 14 BPTI bound cases and the individual 1BTH repeat and unbound 9BTI cases. The black line shows the standard deviation across the 14 receptor complexes; thus, the midpoint of each black line denotes $F_{\text{D,avg.}}$ for that residue. The unbound ligand $F_{\text{D,9PTI}}$ is shown as a circle, while the 1BTH repeat $F_{\text{D,1BTH}}$ is a cross. The method by which F_{D}, $F_{\text{D,9PTI}}$, and $F_{\text{D,1BTH}}$ are calculated is provided in Section 2.6.4.2. The secondary structure associated with a specific residue is indicated at the top, with the traditional terms of C (unstructured, white), H (α-helices, purple), and E (β-sheets, yellow) used as labels. The three domains, grey, red and blue associated with each residue as shown in Figure 2.6.1, are indicated by the background colour. **(b)** The unbound BPTI and 1BTH repeat F_{D} data points scaled as multiples of the standard deviation from $F_{\text{D,avg.}}$. The dashed line indicates 3σ from $F_{\text{D,avg.}}$ for that residue.

Looking at Figure 2.6.2(a), we observe that for every residue the 1BTH repeat displays dynamics more similar to the original 1BTH simulation than any other complex. Any difference between these the repeat and the original arises due to a general noise contribution. We note that to some extent, almost every residue, even those distal to the binding site, had their dynamics altered through the BPTI ligand binding to a different receptor than thrombin, although there is no clear means to distinguish which amino acids belong to which domain. At this domain level, the traditional means of measuring structural/dynamic changes, the RMSD, yields little information; with the RMSD averaged over the three-domain residues between each complex and the 1BTH benchmark: grey (0.8(4) Å), red (0.8(5) Å), and blue (0.7(4) Å), being roughly equal. In contrast, $F_{\text{D,avg.}}$ over the domains, corrected for the noise found through the 1BTH repeat, has identified changes; with the grey (0.078(2)) and red (0.052(2)) each showing levels of impact that are less than the binding site blue domain (0.088(3)). Thus, while we are not able to identify which residues belong to which domain, we can identify specific changes in behaviour due to distal binding on a domain-level scale.

Exploring the impact on residue-level dynamics, Figure 2.6.2(b) allows us to determine which residues are generally impacted by the formation of the 1BTH complex. Residues below the dotted line fall within a 3σ range of $F_{\text{D,avg.}}$. In other words, these residues did not show any significant structural or conformational change from the other complexes outside of the general noise from the simulation. Residues that fall outside this range had their conformational dynamics fundamentally altered due to binding to the thrombin receptor. Table 2.6.2 provides information on these residues.

Table 2.6.2: Residues that had a noteable change in registered dynamics through BPTI binding to thrombin in 1BTH relative to the other 14 complexes. Reported are the number of standard deviations each 1BTH BPTI quantity is away from the average of the other 14 complexes, hence quantities > 3 are of particular interest. Specifically, we report on the deviation of: (1) $F_{\text{D,1BTH}}$ from $F_{\text{D,avg.}}$, (2) the time-averaged RMSD of the 1BTH repeat from the other 14 case's time-averaged RMSD, and (3) the single-frame RMSD equivalent (see Section 2.6.4.2).

Residue	Domain	$F_{\text{D,1BTH}}$	Time-Averaged RMSD	Single-Frame RMSD
R1	Grey	9.35	6.01	1.51
R20	Blue	4.33	6.36	1.81
G37	Blue	3.41	0.77	1.19
C38	Blue	4.75	0.44	2.37
R42	Red	12.20	3.44	0.30
G56	Grey	3.75	1.14	1.18

The RMSD is calculated between each BPTI flavour and the first frame of the 1BTH reference. As a fair comparison to our fraction of dynamics that are different method, we averaged the RMSD difference, across every 5 ps frame in 600 ps, between each residue and the respective residue from the first frame of the 1BTH reference simulation, before reporting on the average across the 14 complexes. This is effectively an RMSF of the trajectory that is then compared to the reference structure. In reality, most studies would report a single-frame RMSD, *i.e.* a comparison between two singular structures; hence we also report the single-frame RMSD for completeness. These RMSD ratios are calculated similarly to the F_{D} multiple, and are described in more detail in Section 2.6.4.2. Clearly, the average RMSD gives a better idea as to which residues are impacted by binding relative to the single frame RMSD. However, even the more informative average RMSD fails to identify three residues: G37, C38 and G56, which have had their native dynamics significantly altered.

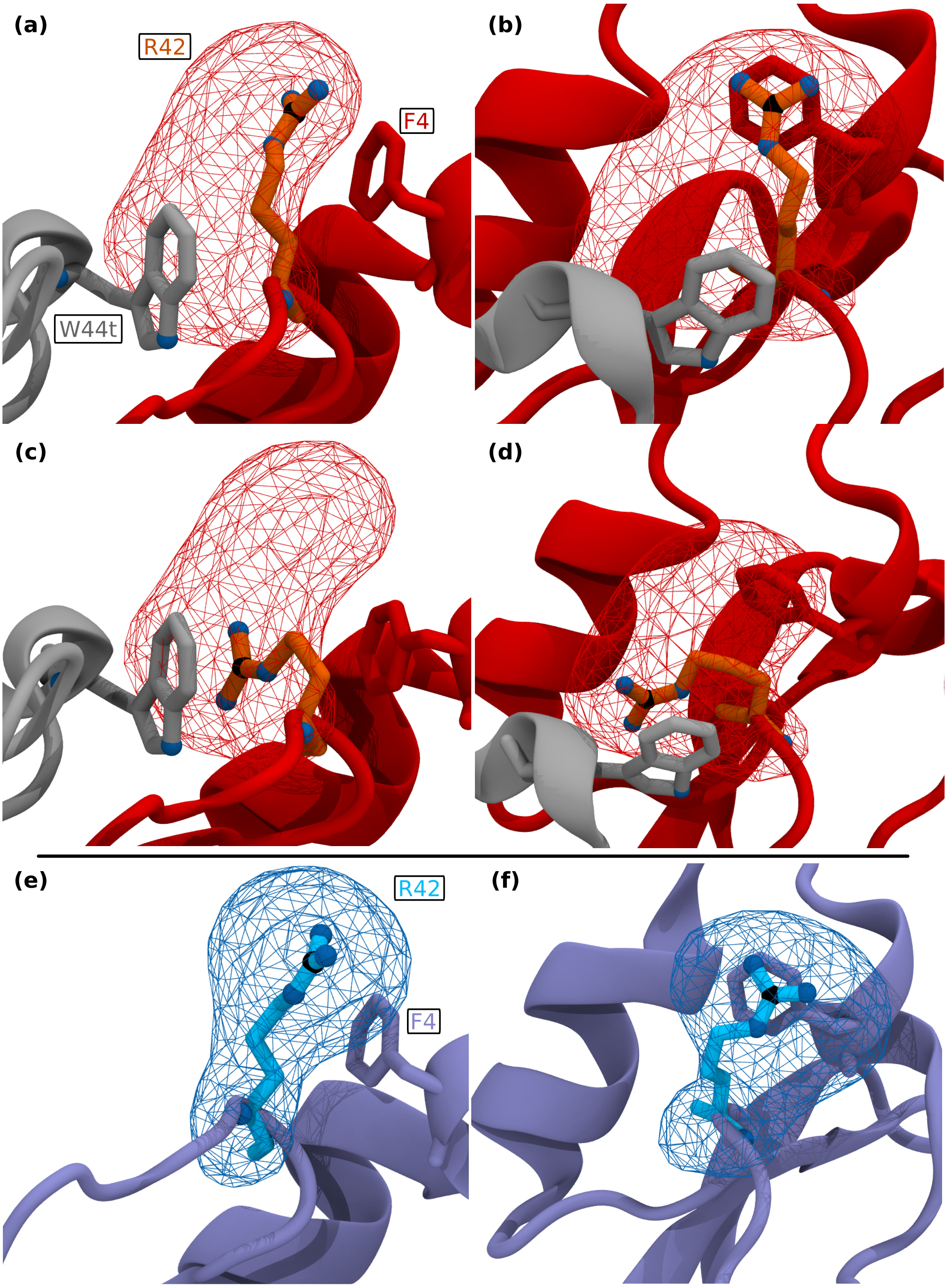

Figure 2.6.3: **(a-b)** 1BTH BPTI (red) R42 (orange) forming a cation-π bond with F4 in the same 1BTH BPTI, and **(c-d)** 1BTH BPTI R42 forming a cation-π bond with 1BTH thrombin (grey) W44t, the t referring thrombin. **(e-f)** 1BZX BPTI (blue) R42 (cyan) exclusively interacting with 1BZX BPTI F4. The wireframes indicate the STID map of R42 at a cutoff of 0.43.

Examining the cause of these dynamics fraction and time-averaged RMSD changes, we observe that for R20 and R42, the changes are all due to specific interactions with the thrombin receptor at the binding site which are not present in the other complexes. R20 is affected by the formation of a salt bridge with thombin's E39 and D60. Taking R42 as a visual example (see Figure 2.6.3), we see that in the majority of BPTI complexes, R42 is stabilised *via* a cation-π interaction with F4 (Figure 2.6.3(e-f)). However, in 1BTH, the thrombin receptor presents a neighbouring tryptophan to interact with. It has been shown that, among the aromatics, tryptophan hosts the most energetically favourable cation-π interaction[67] with arginine. Thus, R42 spends the majority of its time in a conformational state that facilitates a cation-π interaction with thrombin's W44, but due to thermal fluctuations is still able to switch states and interact with F4 in BPTI, hence the spread of dynamics shown in Figure 2.6.3(a-d). The single-frame RMSD comparison is unable to capture this switch.

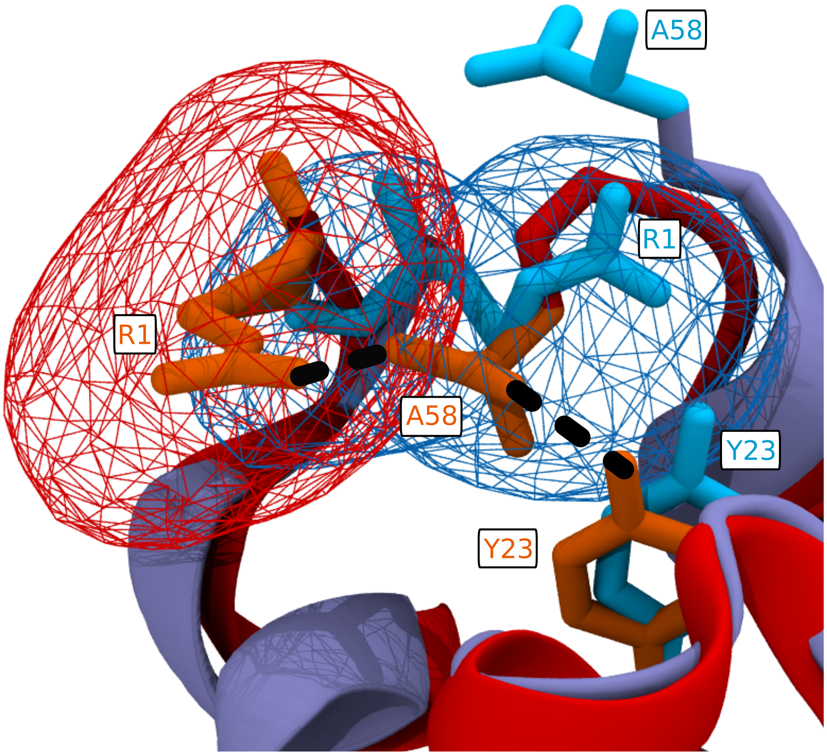

Figure 2.6.4: Hydrogen bonding network between Y23>A58>R1 in 1BTH (red, specific residue in orange) BPTI, faciliated by R42 not forming a cation-π bond with F4. The equivalent residues are indicated for 1BZX (blue, specific residue in cyan) BPTI. R1 and A58 may still form a hydrogen bond in 1BZX, but the formation of the above indicated H-bonding network predicates the Y23>A58 interaction. The wireframe indicates the STID map of R1 at a cutoff of 0.43.

These changes to the occupied dihedral conformational states of both R20 and R42 disrupts the internal hydrogen bonding network of BPTI (see Section 2.6.4.3 for how this network is predicted). Retaking R42 as an example, the cation-π interaction with F4 promotes a hydrogen bond between F4 and E7. However, the presence of thrombin's W44 significantly reduces the probability that this interaction is occupied; 1% in 1BTH *versus* 20% averaged over the other complexes, measured over 600 ps. In turn, this frees up E7 to interact with N43, and subsequently facilitates a H-bonding network; E7>N43>Y23>A58>R1, that results in R1 being pulled towards the centre of the BPTI ligand as opposed to open to the solvent (see Figure 2.6.4). This 'closed'

R1 state is further stabilised *via* the formation of a salt bridge between R1 and D3. Since R1 in 1BTH spends a significant amount of time in this closed state, it is not able to form a hydrogen bond with G56, thereby inducing a further dynamics shift.

2.6.3 Discussion

The average RMSD was able to identify a dynamics shifts for R1, R20 and R42. Note that R1, as a terminus, will naturally have a larger RMSD – but these internal fluctuations are accounted for in the reported ratio. All three are large arginines, where the all-atom time-averaged RMSD can easily identify shifts between dihedral states. The larger dynamics fraction change ratio also reflects this fact. However, the time-averaged RMSD method struggles with the smaller residues' more subtle dynamic changes, wherein G37, C38 and G56 had their conformational space altered *via* the reshuffling of BPTI's internal H-bonding network. The RMSD, calculated as an average of all atoms in a residue, is weighted towards backbone atoms in these smaller residues, in contrast to the side-chain atoms having a greater weighting in arginine. Therefore, because the overall backbone motion is comparable between the BPTIs, these residues seemingly behave similarly if we only look to the RMSD as a means of comparison. In contrast, our STID map based dynamics fraction method clearly identified these smaller residues as of interest to a study into the impact of binding on BPTI.

We have compared individual residue STID maps for BPTI extracted from 1BTH with 14 different BPTI bound complexes, such that we could determine how the thrombin receptor affects the dynamics of individual residues. We have shown that while our STID map-based method does not directly provide allosteric information, it does yield information on which residue's local structure and dynamics are impacted by a binding event. This made possible an additional investigation into the H-bonding network (see Section 2.6.4.3), which provided information on how residue dynamics both proximal and distal from the binding site are linked. In particular, our STID map method identified subtle residue conformational changes not found through RMSD structural comparison methods. We note that, due to the nature of the method, it is not applicable when larger allosteric or domain-level conformational changes occur, although these are usually easier to distinguish. Instead, the method's scope is to explore the impact of some distal binding event on individual residues that sample their local space over short, picosecond timescales. In contrast to Ho and Hamelberg's method,[64] it can also be scaled to substantially larger proteins, with simulation and STID map build times taking only a few minutes on decent hardware. For practicality in such a scenario, we recommend comparing the bound complex of interest with several repeats of either the unbound ligand or another bound complex.

2.6.4 Methods

2.6.4.1 Molecular Dynamics

See Section 2.3.5.1 for details on the MD setup used for this work.

2.6.4.2 Generating the dynamics fraction & RMSD ratio quantities

Following extraction of the BPTI simulation from each host complex, we aligned each simulation along the backbone such that the RMSD between them was minimised. When building the STID map for each residue, we ensured that each map's origin was in the same position and that the generated meshgrid had the same length along each Cartesian axis. Each voxel is kept at 1 Å^3 in volume. The result of these STID maps, at a cutoff of 0.43 (determined through our benchmark in Section 2.3.4), can be seen in Figures 2.6.4 & 2.6.3.

When comparing STID maps to consider the fraction of dynamics that are different, F_{D}, we first convert each STID map into a signed distance field (SDF), defining the zero contour at a value of 0.43. This means that rather than each voxel in space corresponding to a STID map value, it instead returns the distance to the surface, by definition the same shape as the isosurfaces previously described. Points outside the surface have an associated negative value, points inside a positive value. Further details on SDFs are given in Section 4.4.1. To define the 'volume' of space that is occupied differently by the dynamics of a residue and the equivalent 1BTH BPTI reference residue, we calculate the inverse union between the SDFs of these two residues: D_1 and D_2, through the union minus the intersection:

$$\begin{aligned} V_{1,\text{I}} &= \max\{-D_1, D_2\}, \\ &= D_1 - D_1 \cap D_2 = \{x : x \in D_1 - x \in D_1 \text{ and } x \in D_2\}, \end{aligned} \tag{2.6.1}$$

where $V_{1,\text{I}}$ represents the SDF of D_1 not also encompassed by D_2. We then apply Equation 2.6.1, swapping the indices, to find $V_{2,\text{I}}$. We can then calculate the SDF that represents the volume occupied by both residues, V_{U}, through the intersection of both SDFs:

$$\begin{aligned} V_{\text{U}} &= \min\{D_1, D_2\}, \\ &= D_1 \cap D_2 = \{x : x \in D_1 \text{ and } x \in D_2\}. \end{aligned} \tag{2.6.2}$$

F_{D}, therefore, is given as a ratio:

$$F_{\mathrm{D}} = \frac{|v_{1,\mathrm{I}}| + |v_{2,\mathrm{I}}|}{|v_{1,\mathrm{I}}| + |v_{2,\mathrm{I}}| + |v_{\mathrm{U}}|}, \tag{2.6.3}$$

The $|\cdot|$ denotes the size of the SDF set. The individual v terms are defined as:

$$\begin{aligned} v_{1,\mathrm{I}} \subset V_{1,\mathrm{I}}\,;\ v_{1,\mathrm{I}} &= \{x \in V_{1,\mathrm{I}} \mid x \leq 0\}, \\ v_{2,\mathrm{I}} \subset V_{2,\mathrm{I}}\,;\ v_{2,\mathrm{I}} &= \{x \in V_{2,\mathrm{I}} \mid x \leq 0\}, \\ v_{\mathrm{U}} \subset V_{\mathrm{U}}\,;\ v_{\mathrm{U}} &= \{x \in V_{\mathrm{U}} \mid x \leq 0\}. \end{aligned}$$

The midpoint of the black lines in Figure 2.6.2(a) refers to the average of F_{D} across the 14 complexes, $F_{\mathrm{D,avg.}}$, while the black line itself represents the standard deviation across these 14. The 1BTH BPTI repeat, where D_1 is extracted from the 1BTH repeat and D_2 the 1BTH reference, is given as a cross. The unbound ligand, where D_1 is extracted from 9PTI, is shown as a circle.

When reporting on the standard deviation multiples used in Figure 2.6.2(b) and Table 2.6.2, which we denote F_{D} multiple in the main text, we use the following to quantify how far out either $F_{\mathrm{D,1BTH}}$ or $F_{\mathrm{D,9PTI}}$ are from $F_{\mathrm{D,avg.}}$:

$$f_{\mathrm{D,x}} = \frac{F_{\mathrm{D,avg.}} - F_{\mathrm{D,x}}}{\sigma_{F_{\mathrm{D,avg.}}}}, \tag{2.6.4}$$

where x refers to either 1BTH or 9PTI. $f_{\mathrm{D,x}}$ represents the number of standard deviations a specific $F_{\mathrm{D,x}}$ is from $F_{\mathrm{D,avg.}}$, and is effectively the number reported in Figure 2.6.2(b) and Table 2.6.2. Note that Table 2.6.2 only contains data from the 1BTH repeat case. We do not take the absolute of $f_{\mathrm{D,1BTH}}$, as a negative value would indicate that the residue in question samples conformational dynamics further from the 1BTH reference than any of the other BPTI complexes, which would warrent further investigation. Unsurprisingly, however, this did not occur for any of the residues.

When calculating the standard deviation multiples for the RMSD method comparison, we use a similar approach. For the time-averaged RMSD, we also consider the contribution of the internal fluctuations that occurs over 600 ps. For this, we first compare, frame by frame, each residue with its equivalent in the first frame of the 1BTH simulation. This returns a distribution of RMSDs for each complexes' BPTI (which we index C) with respect to the 1BTH BPTI reference structure. The associated time-averaged RMSD for each complex is R_C, with standard deviation, σ_{R_C}. We can then calculate an overall time-averaged RMSD, $R_{\mathrm{avg.}}$ across the 14 complexes with a standard deviation representing this distribution, $\sigma_{\mathrm{R_{avg.}}}$. For an application similar to Equation 2.6.4, we must then propogate these different standard deviations, each σ_{R_C} representing the internal fluctuations of a residue, and $\sigma_{\mathrm{R_{avg.}}}$ representing the distribution across the 14 complexes:

$$\sigma_{\mathrm{R}} = \sqrt{\frac{1}{14}\sum_{C=1}^{n=14}\sigma^2_{\mathrm{R}_C} + \sigma^2_{\mathrm{R}_{\mathrm{avg.}}}}, \tag{2.6.5}$$

where σ_{R} is the overall standard deivation for the RMSD fluctuations. Finally, we can calculate the time-averaged RMSD ratio, f_{R}, reported in Table 2.6.2 as a multiple of σ_{R}:

$$f_{\mathrm{R}} = \frac{R_{\mathrm{avg.}} - R_{\mathrm{1BTH}}}{\sigma_{\mathrm{R}}}, \tag{2.6.6}$$

where R_{1BTH} is the average RMSD from the repeat 1BTH BPTI simulation. Note that we do not propagate the standard deviation from the distribution of RMSDs from the 1BTH repeat simulation to avoid a scenario where R_{1BTH} and $R_{\mathrm{avg.}}$, within six effective σ of one another, is registered as a residue unaltered through ligand binding. The single-frame RMSD comparison also uses Equation 2.6.6, with the stipulation that σ_{R} is generated only from $\sigma_{\mathrm{R}_{\mathrm{avg.}}}$, and R_{1BTH} is not an average across a simulation.

2.6.4.3 Predicting the internal H-bonding network

Once residues of interest are identified through the dynamics fraction ratio method, we can then explore which other residues can form H-bonds using the VMD[68] HBonds plugin.[69] For this, we utilised the default settings: a donor–acceptor distance of 3.0 Å, and an angle cutoff of 20°. The plugin reports, for every frame in a simulation, the H-bonds that can form between a residue of interest and the rest of the protein. It subsequently provides each residue that is interacted with and an occupancy level, *i.e.* how much of the simulation is spent in a position where the two are interacting. We can apply this process iteratively to each connected residue in a chain-like manner to provide insight into which residues are linked through a H-bonding network.

2.7 Conclusions

The approximation of treating a protein as a spatially fixed series of atoms connected by rigid bonds can seem adequate for specific applications, such as utilising Voronoi tesselation to yield the volume and, thus, the density of a protein for X-ray crystallography. Yet, critically it surrenders the key characteristic of dynamics, consequently ignoring its influence on said volume. In this Chapter, we have introduced a novel means to represent a protein through the Spatial and Temporal Influence Density map, capable of simultaneously encompassing the electrostatics, shape and dynamics within a single representation.

We have shown that this model is built from well-grounded principles and that the parameters used in generating it have been thoroughly and objectively benchmarked. In other words, the isosurface we obtain emerge naturally from the properties of the STID map, itself a property of simulating a protein in solution (or vacuum), without empirical considerations for later applications.

We have applied our model to three different, distinct physical and biophysical problems. First, we showed that the theory used to build the STID map could also predict dielectric trends for different solvents, albeit with the caveat that there are superior computational methods for measuring small variations in the dielectric properties of a solvent or protein. We then demonstrated that the excluded volume a STID map can be considered to represent, has implications in the widely accepted value for the density of a protein, generating a new relationship based on the dynamics of a protein rather than the mass. This relationship is robust in different environmental conditions and more accurately matches recent experimental work used to measure protein density, with subsequent applications in a wide range of fields, such as the study of amyloid aggregation.[46] Finally, we have shown that STID maps can distinguish local regions of interest on a protein that are, or indeed are not, impacted through the formation of a complex as a preliminary means to identify which amino acid's dynamic conformational states are influenced by some distal binding interaction. This could act as a starting point in fields such as drug discovery.

Therefore, we have established that the STID map is a remarkable model and tool for several research areas, owing to its ability to encapsulate side-chain dynamics, electrostatics and protein shape. These features are essential in modelling how a protein experiences and interacts with its environment, which is demonstrated through arguably the STID map's most important application in the next Chapter's work on protein-protein docking. While all of these characteristics are important, it is the inclusion of dynamics in the model that is novel over other comparative methods, particularly in the arena of protein docking. Therefore, in Chapters 3 & 4, we will largely focus on how better consideration of dynamics yields higher quality solutions in a protein docking context, yet it should be emphasised that all three of dynamics, electrostatics and protein shape, are present and play a role in determining the shape of the STID map, and subsequently its successful application in protein-protein docking.

2.8 Bibliography

[1] Y. Yan and S. Y. Huang, *Pushing the accuracy limit of shape complementarity for protein-protein docking*, *BMC Bioinformatics*, 2019, **20**, 696.

[2] V. I. Lesk and M. J. Sternberg, *3D-Garden: A system for modelling protein-protein complexes based on conformational refinement of ensembles generated with the marching cubes algorithm*, *Bioinformatics*, 2008, **24**, 1137–1144.

[3] Z. Shentu, M. Al Hasan, C. Bystroff and M. J. Zaki, *Context shapes: Efficient complementary shape matching for proteinprotein docking*, *Proteins Struct. Funct. Bioinforma.*, 2008, **70**, 1056–1073.

[4] R. Norel, D. Petrey, H. J. Wolfson and R. Nussinov, *Examination of shape complementarity in docking of unbound proteins*, *Proteins Struct. Funct. Genet.*, 1999, **36**, 307–317.

[5] R. Chen and Z. Weng, *A novel shape complementarity scoring function for protein-protein docking*, *Proteins Struct. Funct. Genet.*, 2003, **51**, 397–408.

[6] W. Wriggers, *Conventions and workflows for using Situs*, *Acta Crystallogr. Sect. D Biol. Crystallogr.*, 2012, **68**, 344–351.

[7] H. Fischer, I. Polikarpov and A. F. Craievich, *Average protein density is a molecular-weight-dependent function.*, *Protein Sci.*, 2004, **13**, 2825–8.

[8] J. G. Kirkwood, *The dielectric polarization of polar liquids*, *J. Chem. Phys.*, 1939, **7**, 911–919.

[9] H. Fröhlich, *Theory of dielectrics: dielectrics constant and dielectric Loss.*, Clarendon Press, Oxford, 1958, vol. 80.

[10] M. Neumann, O. Steinhauser and G. S. Pawley, *Consistent calculation of the static and frequency-dependent dielectric constant in computer simulations*, *Mol. Phys.*, 1984, **52**, 97–113.

[11] J. W. Pitera, M. Falta and W. F. Van Gunsteren, *Dielectric properties of proteins from simulation: The effects of solvent, ligands, pH, and temperature*, *Biophys. J.*, 2001, **80**, 2546–2555.

[12] D. V. Fedorov, M. Sadhukhan, M. Stöhr and A. Tkatchenko, *Quantum-Mechanical Relation between Atomic Dipole Polarizability and the van der Waals Radius*, *Phys. Rev. Lett.*, 2018, **121**, 3401.

[13] T. Vreven, I. H. Moal, A. Vangone, B. G. Pierce, P. L. Kastritis, M. Torchala, R. Chaleil, B. Jiménez-García, P. A. Bates, J. Fernandez-Recio, A. M. Bonvin and Z. Weng, *Updates to the Integrated ProteinProtein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2*, *J. Mol. Biol.*, 2015, **427**, 3031–3041.

[14] K. Henzler-Wildman and D. Kern, *Dynamic personalities of proteins*, *Nature*, 2007, **450**, 964–972.

[15] D. Mikhailov, V. A. Daragan and K. H. Mayo, *Lysine side-chain dynamics derived from 13C-multiplet NMR relaxation studies on di- and tripeptides*, *J. Biomol. NMR*, 1995, **5**, 397–410.

[16] Y. Gu, D. W. Li and R. Brüschweiler, *Decoding the mobility and time scales of protein loops*, *J. Chem. Theory Comput.*, 2015, **11**, 1308–1314.

[17] H. J. Berendsen, D. van der Spoel and R. van Drunen, *GROMACS: A message-passing parallel molecular dynamics implementation*, *Comput. Phys. Commun.*, 1995, **91**, 43–56.

[18] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, *ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB*, *J. Chem. Theory Comput.*, 2015, **11**, 3696–3713.

[19] B. Hess, H. Bekker, H. J. Berendsen and J. G. Fraaije, *LINCS: A Linear Constraint Solver for molecular simulations*, *J. Comput. Chem.*, 1997, **18**, 1463–1472.

[20] R. P. Joosten, F. Long, G. N. Murshudov and A. Perrakis, *The PDB-REDO server for macromolecular structure model optimization*, *IUCrJ*, 2014, **1**, 213–220.

[21] A. Shrake and J. A. Rupley, *Environment and exposure to solvent of protein atoms. Lysozyme and insulin*, *J. Mol. Biol.*, 1973, **79**, 351–71.

[22] H. Liu and B. Dkhil, *Effect of resistivity ratio on energy storage and dielectric relaxation properties of 03 dielectric composites*, *J. Mater. Sci.*, 2017, **52**, 6074–6080.

[23] J. Guo, A. L. Baker, H. Guo, M. Lanagan and C. A. Randall, *Cold sintering process: A new era for ceramic packaging and microwave device development*, *J. Am. Ceram. Soc.*, 2017, **100**, 669–677.

[24] X. Zhang, C. Bontozoglou, E. Chirikhina, M. Lane and P. Xiao, *Capacitive Imaging for Skin Characterizations and Solvent Penetration Measurements*, *Cosmetics*, 2018, **5**, 52.

[25] P. Sikivie, Dark Matter Axions '96, 1996, pp. 543–554.

[26] D. E. Khaled, N. N. Castellano, J. A. Gázquez, A. J. Perea-Moreno and F. Manzano-Agugliaro, *Dielectric spectroscopy in biomaterials: Agrophysics*, 2016, `/pmc/articles/PMC5503049/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC5503049/`.

[27] S. Bone, R. S. Lee and C. E. Hodgson, *Dielectric studies of intermolecular interactions in native DNA*, *Biochim. Biophys. Acta - Gene Struct. Expr.*, 1996, **1306**, 93–97.

[28] R. Hölzel, *Dielectric and dielectrophoretic properties of DNA*, *IET Nanobiotechnology*, 2009, **3**, 28–45.

[29] R. Jin and K. J. Breslauer, *Characterization of the minor groove environment in a drug-DNA complex: Bisbenzimide bound to the poly[d(AT)·poly[d(AT)]duplex*, *Proc. Natl. Acad. Sci. U. S. A.*, 1988, **85**, 8939–8942.

[30] D. A. Barawkar and K. N. Ganesh, *Fluorescent d(CGCGAATTCGCG): Characterization of major groove polarity and study of minor groove interactions through a major groove semantophore conjugate*, *Nucleic Acids Res.*, 1995, **23**, 159–164.

[31] A. Cuervo, P. D. Dans, J. L. Carrascosa, M. Orozco, G. Gomila and L. Fumagalli, *Direct measurement of the dielectric polarization properties of DNA*, *Proc. Natl. Acad. Sci. U. S. A.*, 2014, **111**, E3624.

[32] L. Li, C. Li, Z. Zhang and E. Alexov, *On the dielectric "constant" of proteins: Smooth dielectric function for macromolecular modeling and its implementation in DelPhi*, *J. Chem. Theory Comput.*, 2013, **9**, 2126–2136.

[33] W. L. Jorgensen and J. Tirado-Rives, *The OPLS Potential Functions for Proteins. Energy Minimizations for Crystals of Cyclic Peptides and Crambin*, *J. Am. Chem. Soc.*, 1988, **110**, 1657–1666.

[34] D. A. Pearlman, D. A. Case, J. W. Caldwell, W. S. Ross, T. E. Cheatham, S. DeBolt, D. Ferguson, G. Seibel and P. Kollman, *AMBER, a package of computer programs for applying molecular mechanics, normal mode analysis, molecular dynamics and free energy calculations to simulate the structural and energetic properties of molecules*, *Comput. Phys. Commun.*, 1995, **91**, 1–41.

[35] B. Tong, M. Zhao, Y. Toku, Y. Morita and Y. Ju, *Local permittivity measurement of dielectric materials based on the non-contact force curve of microwave atomic force microscopy*, *Rev. Sci. Instrum.*, 2019, **90**, 033706.

[36] L. Fumagalli, A. Esfandiar, R. Fabregas, S. Hu, P. Ares, A. Janardanan, Q. Yang, B. Radha, T. Taniguchi, K. Watanabe, G. Gomila, K. S. Novoselov and A. K. Geim, *Anomalously low dielectric constant of confined water*, 2018, `http://science.sciencemag.org/`.

[37] Z. Xu, H. H. Luo and D. P. Tieleman, *Modifying the OPLS-AA force field to improve hydration free energies for several amino acid side chains using new atomic charges and an off-plane charge model for aromatic residues*, *J. Comput. Chem.*, 2007, **28**, 689–697.

[38] W. L. Jorgensen and J. Tirado-Rives, *Potential energy functions for atomic-level simulations of water and organic and biomolecular systems*, 2005, `https://academic.oup.com/nar/article/45/W1/W331/3747780`.

[39] L. S. Dodda, J. Z. Vilseck, J. Tirado-Rives and W. L. Jorgensen, *1.14CM1A-LBCC: Localized Bond-Charge Corrected CM1A Charges for Condensed-Phase Simulations*, *J. Phys. Chem. B*, 2017, **121**, 3864–3870.

[40] L. S. Dodda, I. C. De Vaca, J. Tirado-Rives and W. L. Jorgensen, *LigParGen web server: An automatic OPLS-AA parameter generator for organic ligands*, *Nucleic Acids Res.*, 2017, **45**, W331–W336.

[41] A. J. Sweere and J. G. Fraaije, *Accuracy Test of the OPLS-AA Force Field for Calculating Free Energies of Mixing and Comparison with PAC-MAC*, *J. Chem. Theory Comput.*, 2017, **13**, 1911–1923.

[42] C. Caleman, P. J. Van Maaren, M. Hong, J. S. Hub, L. T. Costa and D. Van Der Spoel, *Force field benchmark of organic liquids: Density, enthalpy of vaporization, heat capacities, surface tension, isothermal compressibility, volumetric expansion coefficient, and dielectric constant*, *J. Chem. Theory Comput.*, 2012, **8**, 61–74.

[43] E. Harder, V. M. Anisimov, T. Whitfield, A. D. MacKerell and B. Roux, *Understanding the Dielectric Properties of Liquid Amides from a Polarizable Force Field*, *J. Phys. Chem. B*, 2008, **112**, 3509–3521.

[44] A. Jouyban, S. Soltanpour and H. K. Chan, *A simple relationship between dielectric constant of mixed solvents with solvent composition and temperature*, *Int. J. Pharm.*, 2004, **269**, 353–360.

[45] P. Tan, J. Li and L. Hong, *Statistical properties for diffusive motion of hydration water on protein surface*, *Phys. B Condens. Matter*, 2019, **562**, 1–5.

[46] E. Zurlo, I. Gorroño Bikandi, N. J. Meeuwenoord, D. V. Filippov and M. Huber, *Tracking amyloid oligomerization with monomer resolution using a 13-amino acid peptide with a backbone-fixed spin label*, *Phys. Chem. Chem. Phys.*, 2019, **21**, 25187–25195.

[47] K. M. Andersson and S. Hovmöller, *The average atomic volume and density of proteins*, *Zeitschrift fur Krist. - New Cryst. Struct.*, 1998, **213**, 369–373.

[48] J. Tsai, R. Taylor, C. Chothia and M. Gerstein, *The packing density in proteins: Standard radii and volumes*, *J. Mol. Biol.*, 1999, **290**, 253–266.

[49] M. L. Quillin and B. W. Matthews, *Accurate calculation of the density of proteins*, *Acta Crystallogr. Sect. D Biol. Crystallogr.*, 2000, **56**, 791–794.

[50] A. A. Ashkarran, K. S. Suslick and M. Mahmoudi, *Magnetically Levitated Plasma Proteins*, *Anal. Chem.*, 2020, **92**, 1663–1668.

[51] K. A. Mirica, S. T. Phillips, S. S. Shevkoplyas and G. M. Whitesides, *Using magnetic levitation to distinguish atomic-level differences in chemical composition of polymers, and to monitor chemical reactions on solid supports*, *J. Am. Chem. Soc.*, 2008, **130**, 17678–17680.

[52] T. Mandl, C. Östlin, I. E. Dawod, M. N. Brodmerkel, E. G. Marklund, A. V. Martin, N. Timneanu and C. Caleman, *Structural Heterogeneity in Single Particleimaging Using X-ray Lasers*, *J. Phys. Chem. Lett.*, 2020, acs.jpclett.0c01144.

[53] E. G. Marklund, T. Ekeberg, M. Moog, J. L. Benesch and C. Caleman, *Controlling Protein Orientation in Vacuum Using Electric Fields*, *J. Phys. Chem. Lett.*, 2017, **8**, 4540–4544.

[54] F. Eisenhaber, P. Lijnzaad, P. Argos, C. Sander and M. Scharf, *The double cubic lattice method: Efficient approaches to numerical integration of surface area and volume and to dot surface contouring of molecular assemblies*, *J. Comput. Chem.*, 1995, **16**, 273–284.

[55] M. Hutt, T. Kulschewski and J. Pleiss, *Molecular modelling of the mass density of single proteins*, *J. Biomol. Struct. Dyn.*, 2012, **30**, 318–327.

[56] C. R. Chen and G. I. Makhatadze, *ProteinVolume: Calculating molecular van der Waals and void volumes in proteins*, *BMC Bioinformatics*, 2015, **16**, 101.

[57] P. G. Squire and M. E. Himmel, *Hydrodynamics and protein hydration*, *Arch. Biochem. Biophys.*, 1979, **196**, 165–177.

[58] K. Gekko and H. Noguchi, *Compressibility of globular proteins in water at 25C*, *J. Phys. Chem.*, 1979, **83**, 2706–2714.

[59] D. S. Kim, J. K. Kim, C. I. Won, C. M. Kim, J. Y. Park and J. Bhak, *Sphericity of a protein via the β-complex*, *J. Mol. Graph. Model.*, 2010, **28**, 636–649.

[60] A. Maißer, V. Premnath, A. Ghosh, T. A. Nguyen, M. Attoui and C. J. Hogan, *Determination of gas phase protein ion densities via ion mobility analysis with charge reduction*, *Phys. Chem. Chem. Phys.*, 2011, **13**, 21630–21641.

[61] L. Saelices, L. M. Johnson, W. Y. Liang, M. R. Sawaya, D. Cascio, P. Ruchala, J. Whitelegge, L. Jiang, R. Riek and D. S. Eisenberg, *Uncovering the mechanism of aggregation of human transthyretin*, *J. Biol. Chem.*, 2015, **290**, 28932–28943.

[62] P. Zhou, X. L. Yang, X. G. Wang, B. Hu, L. Zhang, W. Zhang, H. R. Si, Y. Zhu, B. Li, C. L. Huang, H. D. Chen, J. Chen, Y. Luo, H. Guo, R. D. Jiang, M. Q. Liu, Y. Chen, X. R. Shen, X. Wang, X. S. Zheng, K. Zhao, Q. J. Chen, F. Deng, L. L. Liu, B. Yan, F. X. Zhan, Y. Y. Wang, G. F. Xiao and Z. L. Shi, *A pneumonia outbreak associated with a new coronavirus of probable bat origin*, *Nature*, 2020, **579**, 270–273.

[63] Y. Cai, J. Zhang, T. Xiao, H. Peng, S. M. Sterling, R. M. Walsh, S. Rawson, S. Rits-Volloch and B. Chen, *Distinct conformational states of SARS-CoV-2 spike protein*, *Science (80-.).*, 2020, **369**, 1586–1592.

[64] K. C. Ho and D. Hamelberg, *Automatic Partition of Protein Molecular Dynamics using Coupled Hidden Markov-Ising Models*, *Biophys. J.*, 2020, **118**, 143a.

[65] D. E. Shaw, P. Maragakis, K. Lindorff-Larsen, S. Piana, R. O. Dror, M. P. Eastwood, J. A. Bank, J. M. Jumper, J. K. Salmon, Y. Shan and W. Wriggers, *Atomic-level characterization of the structural dynamics of proteins*, *Science (80-.).*, 2010, **330**, 341–346.

[66] A. Van De Locht, W. Bode, R. Huber, B. F. Le Bonniec, S. R. Stone, C. T. Esmon and M. T. Stubbs, *The thrombin E192Q-BPTI complex reveals gross structural rearrangements: Implications for the interaction with antithrombin and thrombomodulin*, *EMBO J.*, 1997, **16**, 2977–2984.

[67] J. P. Gallivan and D. A. Dougherty, *Cation-π interactions in structural biology*, *Proc. Natl. Acad. Sci. U. S. A.*, 1999, **96**, 9459–9464.

[68] W. Humphrey, A. Dalke and K. Schulten, *VMD - Visual Molecular Dynamics*, *J. Mol. Graph.*, 1996, **14**, 33–38.

[69] J. Gumbart and D. Luo, *HBonds Plugin*, 2012.

3 JabberDock: Accounting for side-chain dynamics in protein docking

3.1 Introduction

As discussed in Chapter 1, protein-protein docking is a powerful computational tool used to predict the quaternary structure of a protein complex. It is a path to rationalising a protein's function in the body and is used to aid drug design to prevent malfunction. Protein-protein docking is a highly complex optimisation problem that struggles to consider the impact of side-chain packing at an interface. Overall, two approaches are used to describe amino acid side-chains at an interface. They are either represented explicitly, thus requiring the docking method to determine their correct packing, or their presence is described through pseudo-coarse-grained representations. The first method requires highly sophisticated optimisation procedures that may still yield suboptimal arrangements. In contrast, the second method usually ignores the uncertainty in the position of the side-chains that comes naturally with any time dynamics. However, since these pseudo-coarse-grained representations are built from static structures, converting them back into atomic models following docking can result in unfeasible predictions.

Herein, we describe how we can apply our pseudo-coarse-grained new protein volumetric representation, named spatial and temporal influence density (STID) maps in a protein-protein docking context, as a substitute for traditional representations based on static structures. As shown in Chapter 2, isosurfaces derived from STID maps are capable of simultaneously describing protein shape, electrostatics, and local dynamics. We will now demonstrate that the complementarity of these isosurfaces can help identify suitable solutions in a protein-protein docking scenario. While surface complementarity techniques have been used for many years,[1] they are primarily applied in the context of the static protein structure shape representations. Thus, the isosurfaces extracted from STID maps are an attractive alternative to any surface method used to date. STID maps inherently consider any side-chain flexibility by using the general motion of atomic point charges in time as a building block for the model. Indeed, we will demonstrate that a key consequence of this is the retention of accuracy between identified docking models regardless of their difficulty.

Our STID map-based scoring method is embodied in JabberDock, a blind protein docking engine that is supported by the POW^{er} optimisation environment.[2] POW^{er} features a modified Particle Swarm Optimisation (PSO) approach, explicitly adapted to prevent premature convergence and maximise the diversity of solutions. Hereafter, we first outline the scoring function used to build the Potential Energy Surface (PES) associated with the arrangement of two protein STID isosurfaces in Cartesian space. Then, we present a set of benchmarks aimed at testing the accuracy of our protein-protein docking method in line with the CAPRI blind docking competition guidelines.[3] In Section 3.3 we apply JabberDock both to a large dataset of soluble proteins, and, using the same scoring function, a smaller benchmark of integral membrane proteins in Section 3.4, an area with a limited number of software options. Results demonstrate that JabberDock can return models matching the quality of top *de novo* docking algorithms currently available. We will finish by discussing JabberDock's effectiveness when applied to two docking applications, one water-soluble and the other membrane-based.

3.2 The JabberDock protein docking engine

3.2.1 Methodology

JabberDock uses a surface complementarity assessment that takes advantage of an isosurface generated from a STID map at a cut-off of 0.43 (see Section 2.3.4) to construct the PES explored by the PSO algorithm implemented in the POW^{er} optimisation engine. We utilise the CAPRI criteria[3] to both assess case difficulty and the accuracy of a complex predicted by JabberDock.

3.2.1.1 Search space exploration

JabberDock uses a 7-dimensional search space to make implementation easier when roto-translating the STID maps: three dimensions define ligand translation in the Cartesian space, three dimensions define an axis of rotation for this ligand, and one dimension defines a rotation angle around this axis.

In order to navigate the PES associated with the scoring function (see Section 3.2.1.2) and produce an ensemble of possible docked poses, JabberDock leverages a distributed heuristic global optimisation algorithm featured in the POW^{er} optimisation environment — PSO "kick and reseed" (PSO-KaR).[2] PSO-KaR is used to explore the PES over 300 iterations using 80 randomly initialised agents ("particles"). According to the "kick and reseed" procedure, particles converging to a local minimum (*i.e.*, with a velocity decaying to less than 4% of the search space dimension in each direction) were randomly restarted, and a repulsion potential placed at their convergence location. These potentials may generate novel, small minima in the PES, however in general only 10~15 are added, and their measured impact has been found to be negligible in terms of providing false solutions.[4] The whole optimisation process is repeated three times, with the memory of previous repulsion potentials retained from one repetition to the next. In sum, this docking procedure requires the evaluation of 72 000 docking poses.

To obtain a diverse ensemble of solutions, 300 poses are finally selected as representatives from the pool of poses having a positive score using a K-means clustering algorithm on the seven-dimensional coordinates associated with each model.

3.2.1.2 Scoring function

JabberDock's scoring function is based around maximising surface complementarity of an isosurface derived from our representations. Surface complementarity is a fairly generic concept which refers to the maximum amount two shapes can "match" one another, while featuring minimal shape intersection. For a simple example, consider two 3D cuboids with lengths: $a > b >> c$, and ignore the presence of sharp edges or vertices, which are not present in our protein representations. The two cuboids have maximum surface complementarity when the two $a \times b$ faces are perfectly aligned, and minimum complementarity, while maintaining contact, if either of the two have their $b \times c$ faces in contact with the other. In contrast, two perfect spheres will feature their maximum and minimum surface complementarity simultaneously at any geometric orientation. The surface complementarity, S, of any two surfaces can be calculated using:

$$S = \frac{\left\{S_{\mathrm{AB}}\right\}\dfrac{\nu_{\mathrm{A}}}{V_{\mathrm{A}}} + \left\{S_{\mathrm{BA}}\right\}\dfrac{\nu_{\mathrm{B}}}{V_{\mathrm{B}}}}{2}, \tag{3.2.1}$$

where the curly brackets indicate that we used the median of the scores for protein A into B and *vice versa*, ν_{A} is the total number of successful contact points on A in contact with B inside some arbitrary distance cut-off, while V_{A} is the total number of points describing the surface of A. The same is true but for B with the ν_{B} and V_{B} terms. The individual scores are given by:

$$S_{\mathrm{A}_i\mathrm{B}_j} = (\vec{n}_{\mathrm{A}_i} \cdot \vec{n}_{\mathrm{B}_j}{}')\mathrm{exp}\left(-w \mid \vec{x}_{\mathrm{A}_i} - \vec{x}_{\mathrm{B}_j}{}' \mid^2\right), \tag{3.2.2}$$

where $\vec{n}_{\mathrm{A}_i}$ is the normal from a point of interest, i, on A's surface and $\vec{n}_{\mathrm{B}_j}{}'$ the antinormal from the corresponding, nearest point, j, on B. w (0.5 Å^{-2}) is an arbitrary weighting found by Lawrence & Coleman[5] and $\mid \vec{x}_{\mathrm{A}_i} - \vec{x}_{\mathrm{B}_j}{}' \mid$ is the physical distance between the two points. S_{AB} is, therefore, the exhaustive collection of individual $S_{\mathrm{A}_i\mathrm{B}_j}$ for all points on A in contact with B, as defined by our cut-off distance. The arbitrary distance cut-off used to consider contact points was empirically determined to maximise S at 1.6 Å (see Figures 3.2.2 and 3.2.3). The ν_{x} and V_{x} ratio is a term we include to avoid minor contact points providing good scores. Figure 3.2.1 provides a schematic as to how the scoring function is calculated for the case of TEM-1 β-lactamase complexed with its inhibitor (PDB: 1JTG).

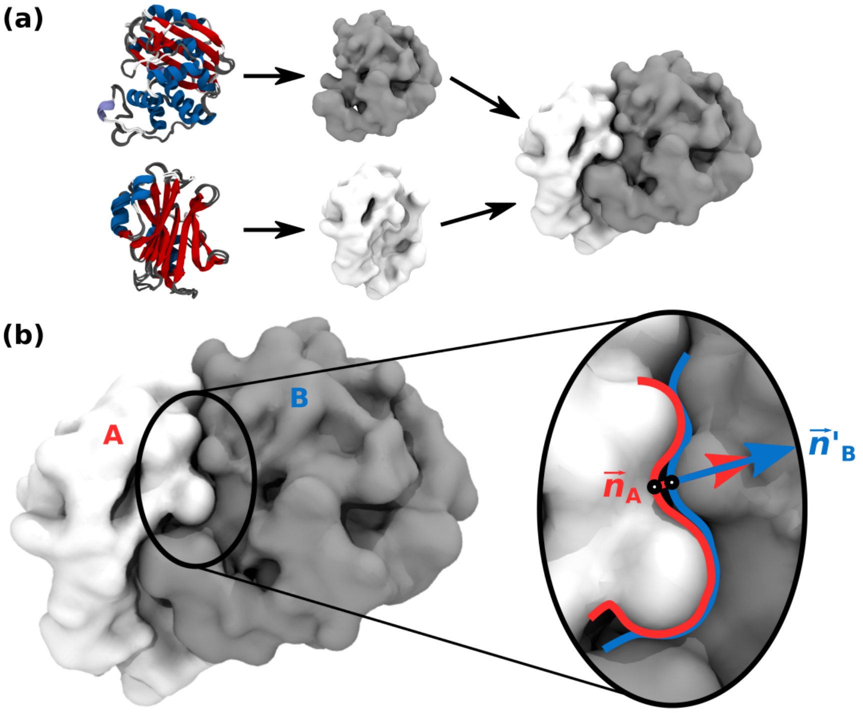

Figure 3.2.1: **(a)** Prediction of TEM-1 β-lactamase complexed with its inhibitor (PDB: 1JTG), showing the protein docking process from first an MD simulation of the monomer units, followed by generation of the STID maps before predictions can be made *via* our scoring function. The combined complex corresponds to the known bound docked position. **(b)** We exhaustively iterate through all points on the surface of each monomer unit. If a point is in contact with the opposing binding partner, we determine a score for that point through the closest point to it. The median from the collection of scores from all individual contacts is used in Equation 3.2.1. In the inset, we show two points in contact, with their corresponding normals used in Equation 3.2.2.

Following a roto-translation of a model requested by the optimiser, a quick test is first performed to identify poses featuring no contact or unphysical atomic overlaps between the ligand and the receptor. Suitable poses, featuring a negative Lennard-Jones potential between the α carbon atoms of receptor and ligand, are scored according to the surface complementary of the STID isosurfaces following the same roto-translation. The shape of the isosurfaces analysed by JabberDock is determined through an isovalue cut-off, with the appropriate value of 0.43 selected based on the benchmark discussed in Section 2.3.4. Independently, we also wished to verify whether this isovalue holds for protein-protein interactions. Figures 3.2.2 and 3.2.3, demonstrate that this cut-off corresponds to the parameter which returns the highest score at the known docked pose ground truth.

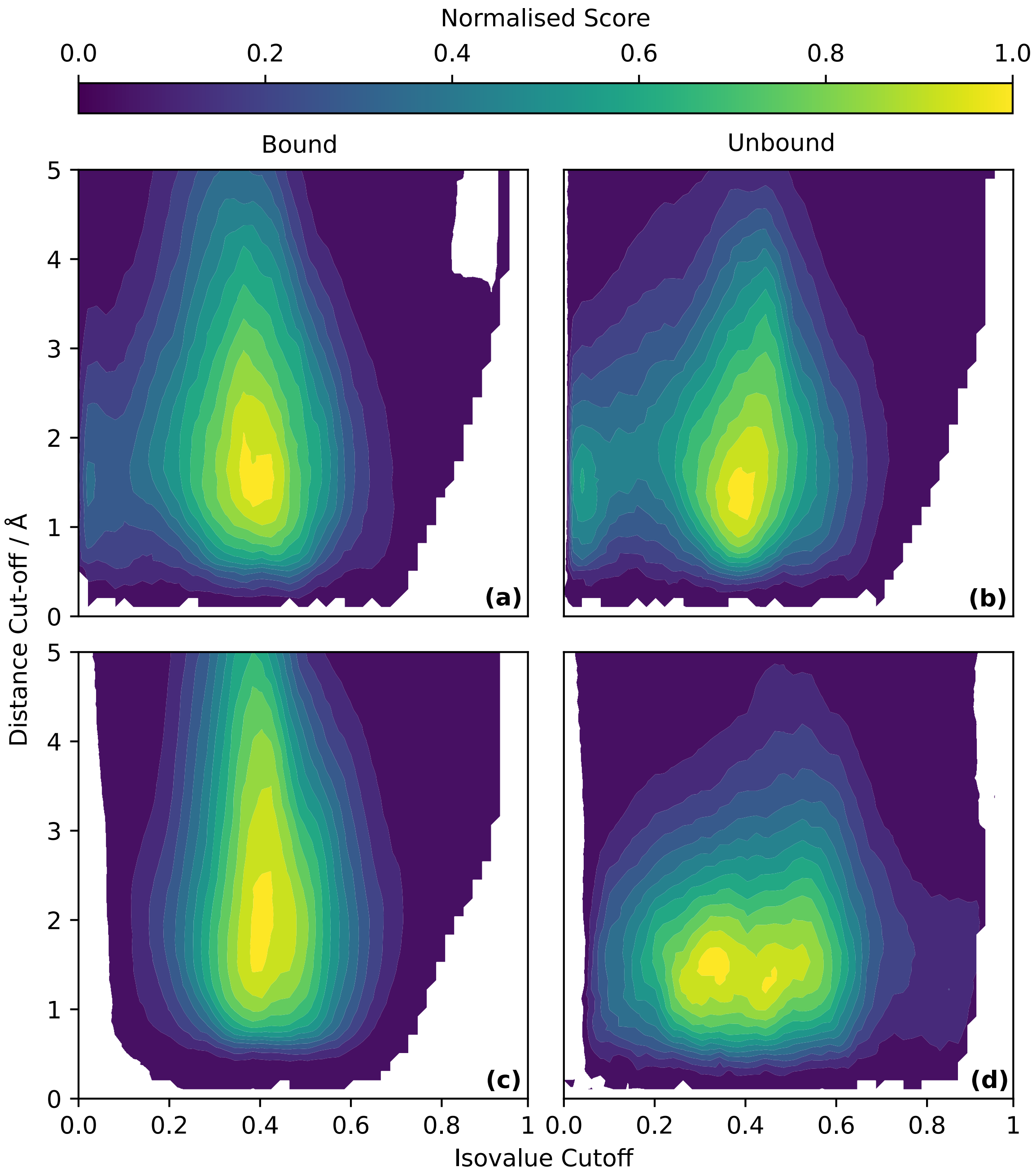

Figure 3.2.2: The score of the known docked pose associated with different combinations of isovalue cut-off and distance cut-off employed to determine if a region of interest is in contact with the surface of the binding partner. The score is normalised independently for each protein. This analysis was run for all our benchmark complexes. This figure shows two example complexes studied both as **(a, c)** bound and **(b, d)** unbound. The complexes are **(a, b)** 1DQJ and **(c, d)** 1PPE. The bound cases show a consistent maximum at 0.43 isovalue and 1.6 Å distance cut-offs. The unbound case of 1DQJ also performs well with these parameters, whereas the unbound 1DQJ case features an additional maximum at an isovalue of 0.3 (see Figure 3.2.3 for a performance comparison of all protein docking cases in the CAPRI benchmark[6]).

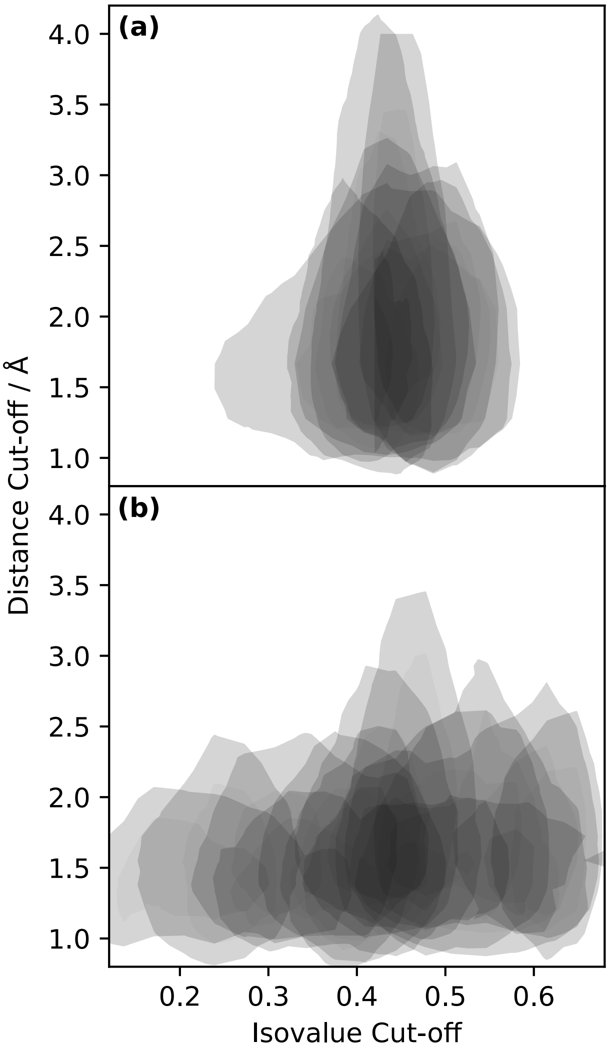

Figure 3.2.3: For each complex in the CAPRI water-soluble protein docking benchmark,[6] we calculated the score of the known docked pose associated with different combinations of isovalue cut-off and distance cut-off. Here, for all **(a)** bound and **(b)** unbound cases we colour the regions of parameter space yielding the highest scores. Two contour levels for each case are shown, representing the top 20% and top 10%. The bound cases clearly show a shared maximum isovalue cut-off of 0.43 and a distance cut-off of 1.6 Å. While the unbound cases are less coherent, with not all cases overlapping, the majority show a maximum at the same parameter choices as the bound cases.

The larger the surface complementary score, the better the fit; thus, the optimiser is set up to maximise the score. Only positive scores are accepted by POW^{er}. Figure 3.2.4 provides three examples of how the raw score for each generated model matches with the metrics used to assess the quality of a solution.

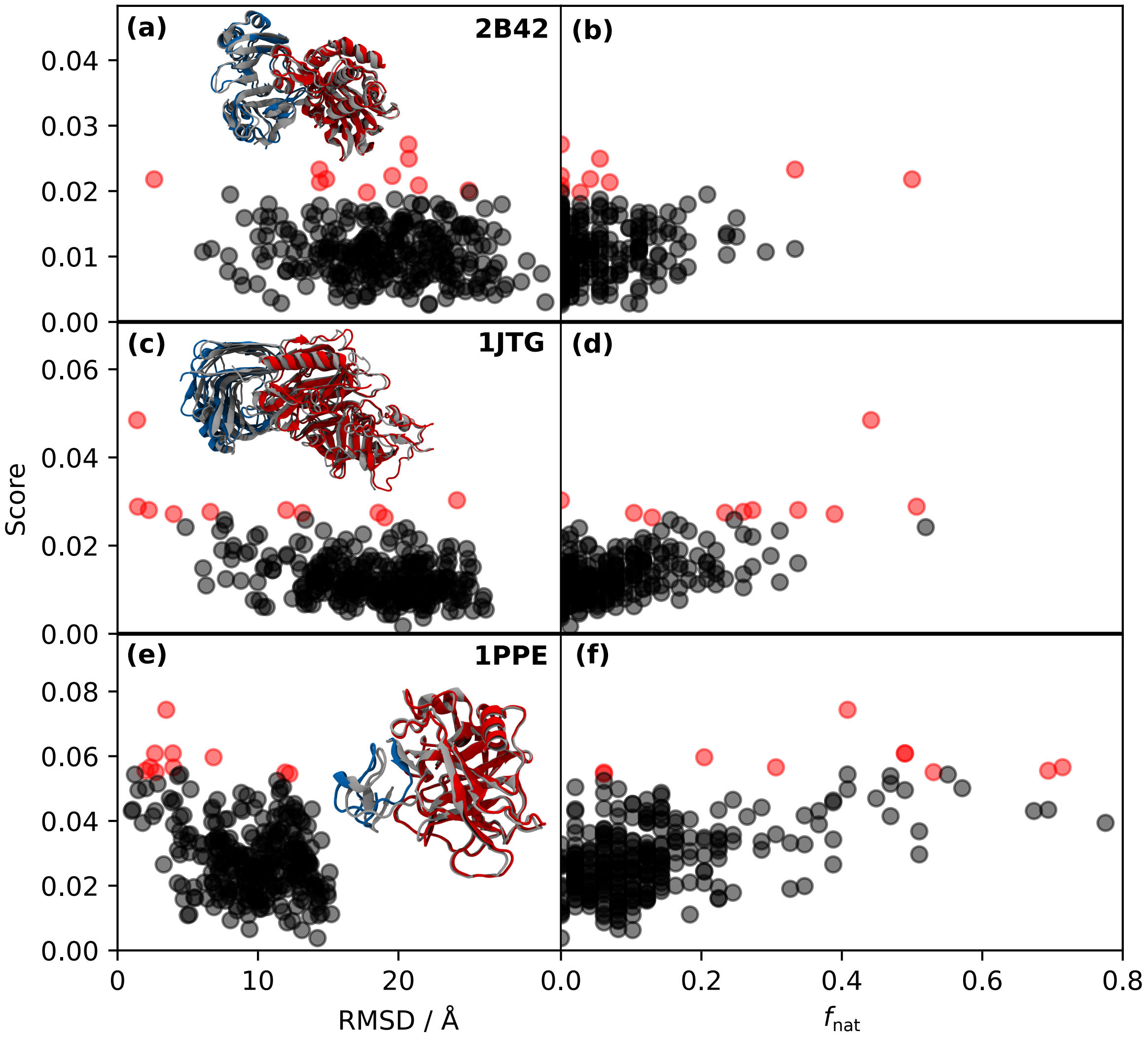

Figure 3.2.4: **(a, c, e)** Variation in overall score with RMSD and **(b, d, f)** Fraction of correctly predicted contact residues ($f_{\mathrm{nat.}}$) for three example unbound cases: **(a, b)** 2B42, **(c, d)** 1JTG, and **(e, f)** 1PPE. The red dots indicate the top 10 docked candidate models that would be submitted under the CAPRI guidelines. An ideal model features a low RMSD and a high $f_{\mathrm{nat.}}$ (see Section 3.2.3.2). The three structures shown correspond to the intermediate quality model in the top 10 ranked solutions predicted by JabberDock. The known bound state (corresponding to the PDB code), is shown in grey, the receptor and ligand used for docking in red and blue respectively.

3.2.2 Implementation

3.2.2.1 Software availability

JabberDock is available for download for free under a GPL license at `github.com/degiacom/JabberDock`. Included is a manual and a tutorial to assist users in setting up JabberDock as well as instructions on how to use it. There are also input and target structures used in our benchmark for the membrane protein docking set, as there is no equivalent benchmark as there is for water-soluble proteins.

3.2.2.2 Computational load

MD simulations were all run with GROMACS[7] 2016, installed with CUDA, on a single GPU using 4 CPUs and 8 OpenMP threads. The specific hardware used included an NVIDIA GeForce GTX 1080 GPU, an Intel®Xeon®CPU E5-2650 v4 @ 2.20 GHz, and 64 GB of DDR4 RAM clocked at 2133 MHz. On average, a complete MD cycle; including building the system, energy minimisation, equilibration, production, and parsing the data, takes 30 minutes for water-soluble proteins, and three days for integral membrane proteins.

The conversion of a trajectory into a STID map can take between 2 and 30 minutes, depending on the size of the system. Each POW$^{\text{er}}$ soluble docking case was run using MPI on 28 Intel®Xeon®CPU E7-8857 v2 @ 3.00 GHz, across 3 TB of memory and a 1× Intel TrueScale 4× QDR single-port InfiniBand interconnect. Figure 3.2.5 shows the relationship between POW$^{\text{er}}$ execution times and the mass of the soluble complex under study. Docking runtimes were faster for membrane proteins, owing to smaller protein sizes and reduced search space. Run across 12 CPUs of the same hardware as above; each docking run took approximately 12 hours. The docking process is sped up by roto-translating the smaller protein ("ligand") and keeping the larger one ("receptor") fixed. The calculation of the number of contact points between two STID maps is the most computationally demanding procedure, acting as a bottleneck unaffected by the choice of ligand and receptor. The RAM usage is substantial, with the largest protein run to date, 1N2C – with a total complex mass of 286.8 kDa, using 65 GB of RAM, the average usage across both soluble and membrane proteins was 10 GB of RAM. We are currently working on improving the processes that have the biggest impact on speed and memory demand.

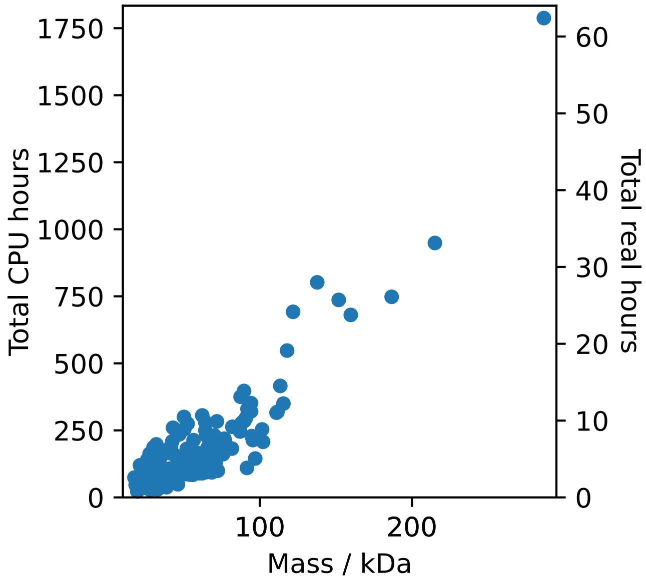

Figure 3.2.5: Relationship between the mass of all water-soluble protein complexes simulated with JabberDock *versus* the time taken to run the entire POW^{er} optimisation search. The runs were made across 28 CPUs, the architecture is discussed in the Computational Load section above. The left y-axis displays the total time to compute per CPU, whereas the right y-axis indicates the real time it took across the 28 CPUs.

3.2.3 Evaluation

3.2.3.1 Case difficulty classification

Protein-protein docking cases are classified under three levels of difficulty, which is associated with their flexibility and the RMSD difference between the Cα atoms at the interface after superposing the bound and unbound interfaces. Cases can be classified as either rigid-body (or easy), medium, or difficult (or hard). Easy cases are those with minimal difference between the unbound crystallised structures and the bound: usually <1 Å difference. In medium cases, the RMSD difference is between 1 Å and ~2.5 Å. Finally, difficult cases can be anything greater than 2.5 Å. Thus, the difficult cases are accordingly significantly more challenging than the other two, particularly given that a difficult case with an RMSD of 10 Å would already lie at the boundary for the requirements of an acceptable success (see Section 3.2.3.2).

3.2.3.2 Assessment of model accuracy

We use three metrics, pre-defined by the CAPRI community,[3] to determine the quality of a model: **(1)** The ratio of correct contact residues (a valid contact defined as an atom within 5 Å of the binding partner) to the number of residues in the predicted complex, $f_{\text{nat.}}$. **(2)** The RMSD between the α-carbons of the known crystal pose and the predicted pose. **(3)** The RMSD of the two poses between the α-carbons at the interface (defined as within 10 Å of the binding partner). This is termed the interfacial RMSD, or RMSD_I. The CAPRI guidelines specify four levels of possible success criteria:

Table 3.2.1: CAPRI criteria conditions of success. RMSD_I stands for interfacial RMSD.

Quality	Conditions to be met
(1) Incorrect	RMSD $>$ 10.0 Å **AND** RMSD_I $>$ 4.0 Å **OR** $f_{\text{nat.}} < 0.1$
(2) Acceptable	RMSD $\leq$ 10.0 Å **OR** RMSD_I $\leq$ 4.0 Å **AND** $0.1 \leq f_{\text{nat.}} < 0.3$ **OR** $f_{\text{nat.}} \geq 0.3$ **AND** RMSD $>$ 5.0 Å **AND** RMSD_I $>$ 2.0 Å
(3) Intermediate	RMSD $\leq$ 5.0 Å **OR** RMSD_I $\leq$ 2.0 Å **AND** $0.3 \leq f_{\text{nat.}} < 0.5$ **OR** $f_{\text{nat.}} \geq 0.5$ **AND** RMSD $>$ 1.0 Å **AND** RMSD_I $>$ 1.0 Å
(4) High	RMSD $\leq$ 1.0 Å **OR** RMSD_I $\leq$ 1.0 Å **AND** $f_{\text{nat}} \geq 0.5$

The protocol for applying this list of inequalities follows the order provided, beginning with defining the incorrect predictions. We qualify a test case as either high, intermediate, or acceptable quality if at least one of its top 10 ranked models matches the criteria above with respect to the target structure.

3.3 Water-soluble protein docking

Author's note: *the raw results reporting on the success of each individual trialled protein case (both for the water-soluble proteins in this Section and the transmembrane protein cases in Section 3.4), are available in a supplementary spreadsheet submitted with this thesis. We refer to this data as Table S1 and S2 for water-soluble and transmembrane protein docking cases, respectively.*

STID maps encapsulate information on the local dynamics of the atomic charges in a protein. This feature is particularly attractive in a protein-protein docking context, as it circumvents the need of determining the specific atomic position of each side-chain at the interface between two binding partners. Therefore, we used this representation within a docking protocol, where the scoring function is determined by the surface complementarity of ligand and receptor STID map isosurfaces (see Section 3.2.1.2). Benefitting from the fact that the structural characteristics reported by each STID map isovalue are protein-independent (see Section 2.3.4), we determined that an isovalue of 0.43 is the most appropriate to report on all electrostatic and dynamic features of any protein within our surface complementarity scheme (see benchmark in Figures 3.2.2 & 3.2.3).

3.3.1 Results

We implemented this calibrated STID map-based scoring into our *de novo* protein docking algorithm: JabberDock. The program utilised the PSO algorithm within the POW$^{\text{er}}$ environment[2] (see Section 3.2.1.1) to explore the energy landscape associated with the arrangement of two binding partners, in search of the arrangement maximising our complementarity score. We assessed the performance of JabberDock against all 230 test cases featured in the most recent iteration of the standard water-soluble protein-protein interaction benchmark[6] (six cases were excluded due to the presence of non-standard amino acids). According to the RMSD between unbound and known bound state, 151 of these cases are classified as rigid-body (easy), 45 as medium, and 34 as difficult (see Section 3.2.3.1). To gather further information on the relationship between docking quality and the conformational change proteins undergo upon binding, we selected a diverse subset of 32 cases (20 easy, 7 medium, and 5 difficult) that were also treated as bound cases. In these cases, the subunits used to predict the assemblies were proteins extracted from the known complex. We classified the quality of all our modelling runs according to the three CAPRI categories: acceptable, intermediate, and high (see Section 3.2.3.2). Hereon, we qualify a test case as a success if at least one model in the top 10 ranked solutions is at least of acceptable quality.

Against the 32 bound cases, JabberDock was successful in 85.0% of the easy, 71.4% of the medium, and 20.0% of the difficult cases. Challenged with the full unbound benchmark set, JabberDock yielded successful predictions for 56.3% of the easy, 60.0% of the medium, and 54.9% of the difficult cases. Although no high-quality predictions were found for any of the test cases, intermediate quality results were found for 29.2% of the easy, 22.2% of the medium, and 25.8% of the difficult cases. Examples of intermediate, acceptable and unacceptable quality structures extracted from the top 10 ranked solutions for three cases of increasing difficulty are provided in Figure 3.3.1. Overall, these results indicate that JabberDock performance is mostly unaffected by the case difficulty (full details are provided in Table S1). These results compare favorably against four of the most commonly used protein-protein docking algorithms: SwarmDock,[8] pyDock,[9] ZDOCK,[10] and HADDOCK.[11] As reported by Vreven *et al.* while setting the benchmark set used in this work,[6] their acceptable success rate for rigid-body cases ranges between 31 and 50%, whereas for the medium and difficult cases, substantially lower success rates (between 4 and 22%) are observed. Regarding intermediate success rates, 13-18% success rates are found. It is only when considering the percentage of high-quality models, where success rates <6% are reported, that JabberDock is outperformed.

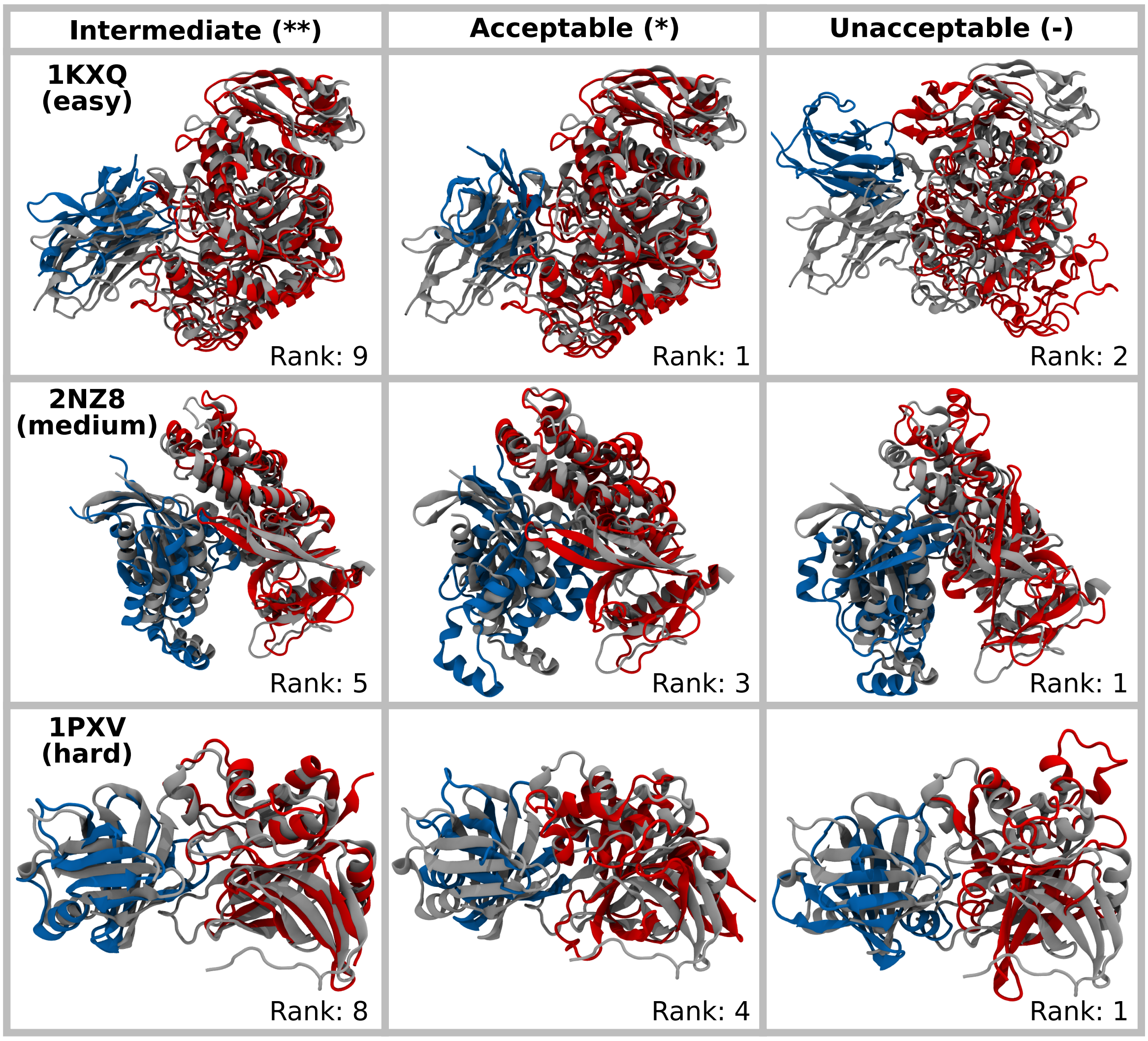

Figure 3.3.1: A selection of top 10 ranked structures generated by JabberDock from three cases of the benchmark, with the ranks indicated bottom right of each subfigure. Specifically, we show a case from the easy dataset: camelid VHH domain complexed with porcine pancreatic α-amylase (PDB: 1KXQ), the medium dataset: N-terminal DHPH cassette of Trio in complex with nucleotide-free Rac1 (PDB: 2NZ8), and the hard dataset: the staphostatin-staphopain complex (PDB: 1PXV). The known bound state, as indicated by the PDB code, is shown in grey, while the receptor and ligand are shown as red and blue, respectively. Note that all three cases would be classified as intermediate successes and that there are many additional unacceptable solutions where the ligand is distal to the binding site, but we show here three cases where it is close for easier comparison. See Table S1 for details.

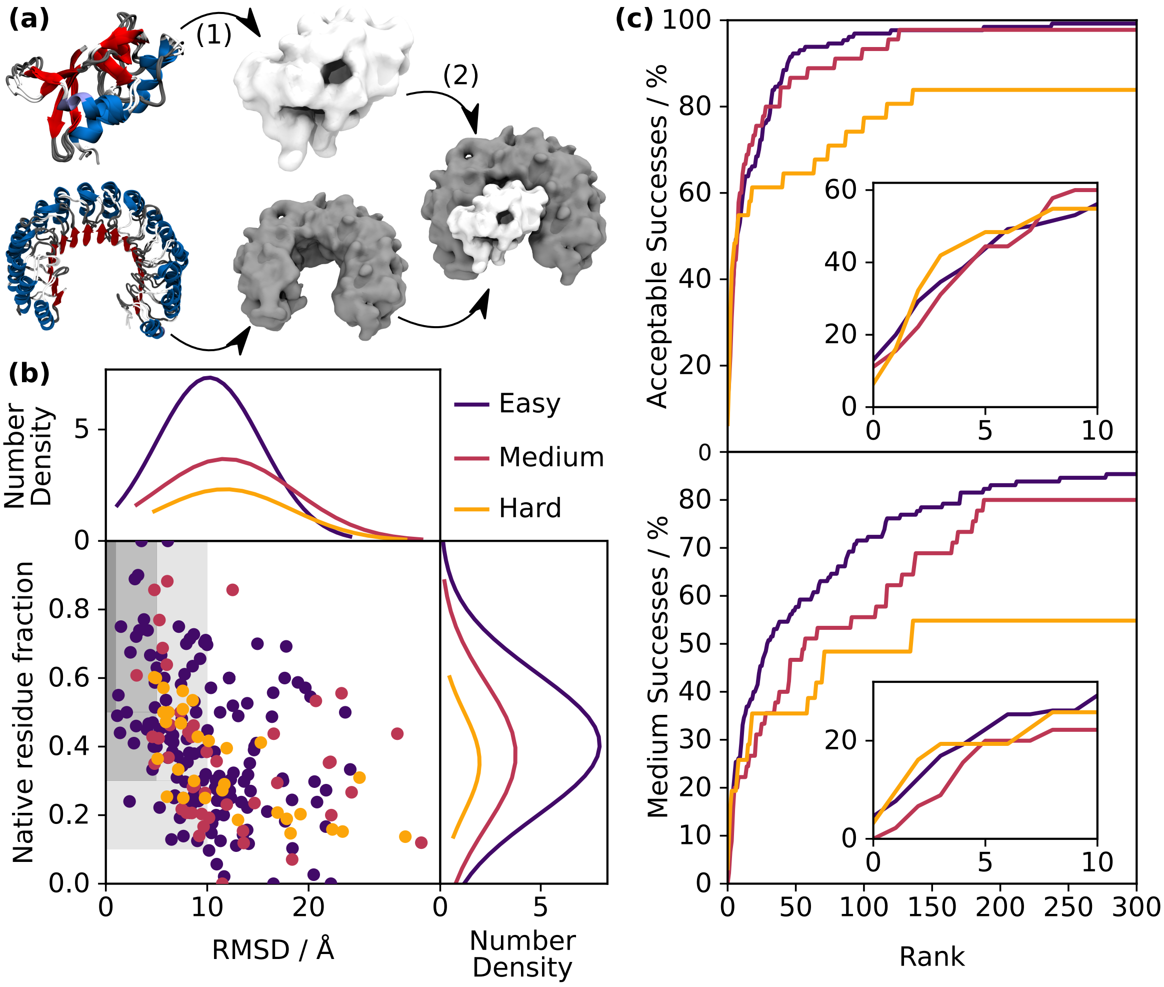

Figure 3.3.2: **(a)** (1) STID map of two binding partners is calculated using their respective MD simulations. (2) STID map representation of both binding partners is leveraged by JabberDock to predict the complex accurately. The image shows the intermediate quality model of ribonuclease A complexed with its inhibitor (PDB: 1DFJ). **(b)** Quality of best models within the top 10 results for every docking case. For each case, the lowest α-carbon RMSD between prediction and crystallised complex is presented, against their associated native residue fraction ($f_{\text{nat.}}$). Point colours indicate the case difficulty, while the dark- to light-shaded regions represent the criteria for high, intermediate, and acceptable quality results, respectively. Thus, a point landing in one of these regions indicates that the corresponding success was found within the top 10 ranked JabberDock solutions. The top and right adjoining subplots show, respectively, the distribution of RMSDs and $f_{\text{nat.}}$ across the models. **(c)** Percentage of test cases yielding an acceptable (top) and intermediate (bottom) success, as a function of the number of ranked structures considered as candidate models. Data are reported independently, in different colours, per case difficulty. The region corresponding to the top 10 models is shaded and magnified in the insets. In this region, JabberDock's success rate is consistent between easy, medium, and difficult docking cases. In the larger pool of 300 models, an acceptable solution is almost always found for the easy cases.

The reason behind JabberDock's consistent performance throughout cases of different difficulties lies in its ability to identify interfacial amino acids correctly. Indeed, while the RMSD of models *versus* the known complex is lower for cases with more flexible subunits, the average ratio of correct contact residues ($f_{\text{nat.}}$) remains nearly unaltered (Figures 3.3.2(b) and 3.3.3). The relationship between the number of candidate models selected from JabberDock's ranked solutions and the resulting success rate features an initial steep gradient (Figure 3.3.2(c)). This indicates that the ranking of JabberDock's first successful model is most likely to be high. Still, by increasing the number of candidate models to 100, results with significantly smaller RMSD and higher $f_{\text{nat.}}$ can be found (Figure 3.3.3). Thus, while most successful models usually rank highly according to our scoring function, better models may well be available when considering a larger pool of solutions. For instance, in 98.6% of easy cases, our full data sets of 300 solutions contained at least one acceptable pose (Figure 3.3.2(c)).

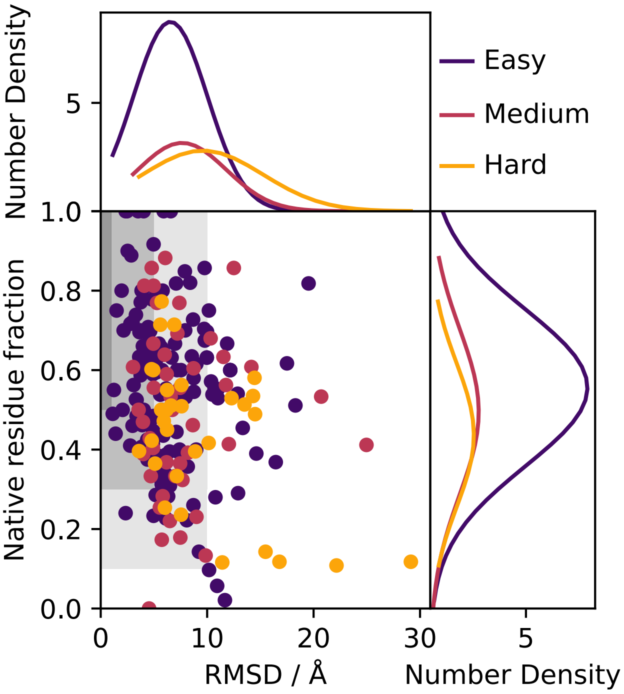

Figure 3.3.3: Quality of best models within the top 100 results for every docking case. For each case, the lowest α-carbon RMSD between prediction and crystallised complex is presented, against their associated native residue fraction ($f_{\text{nat.}}$). Point colours indicate the case difficulty, while the dark to light shaded regions represents the criteria for high, intermediate and acceptable quality results, respectively. Thus, a point landing in one of these regions indicates that the corresponding success was found within the top 100 ranked JabberDock solutions. The top and right adjoining subplots show, respectively, the distribution of RMSDs and $f_{\text{nat.}}$ across the models.

By manually aligning each unbound monomeric subunit to their bound counterpart in the complex and assessing the score achieved at the known docked position, we observed that four of the easy and one of the difficult cases yielded scores higher than anything found by JabberDock (see Table S1). One example is the xyloglucan-specific endo-β-1,4-glucanase (PDB: 3VLB), where the two binding partners are highly interlocked. In this case, failure was not caused by an unsuitable scoring function, but by an underperforming optimiser, which was unable to navigate into the complex binding site. Many of the successful unbound cases feature interlocked arrangements. In such cases, if the optimiser can identify the narrow set of roto-translations allowing the binding partners to interlock, the resulting model will have a high score. A successful example is that of the β-lactamase TEM1 (PDB: 1BTL) — ribonuclease A (PDB: 9RSA) complex, involving a significantly large and complex contact region, whereby almost the entire circumference of β-lactamase's STID map is buried (see Figure 3.3.2(a)).

Unsuccessful cases, such as the profilin-β-actin complex (PDB: 2BTF), most often feature a flat binding site. In these cases, the surface complementarity score alone struggles to discriminate between binding and non-binding regions because of a lack of characteristic surface features, and thus, successful models do not rank highly. Addressing these cases requires capturing additional properties of protein-protein interactions. To this end, we explored the possibility of reranking JabberDock models accounting to the vectoral alignment of neighbouring dipoles (see Figure 2.2.1(b)) at the interface *via* the dipole maps used to build the STID maps (see Section 3.3.3.2). This score is aimed at favouring arrangements featuring alignment of the dipoles. For instance, during hydrogen bond formation the dipoles neighbouring the hydrogen would point toward the hydrogen, whereas those near the oxygen would be pointing away; hence, the two would be roughly aligned within some cut-off. The 2 Å cut-off was chosen based on the principle that there must be contacts within a typical intermolecular bond range. This method serves to penalise arrangements where the dipoles of the two maps are uncorrelated. The dipole score was calculated for each predicted complex and added to the surface complementarity score (Equation 3.2.1) in an unweighted sum. The final scores were then used to determine the overall rank. Results indicate that such a post-processing reranking, while significantly improving the quality of poses for 12% of the data set, overall decreased the success rate. In Section 4.4 we will explore alternative means to accommodate these featureless surfaces.

The most flexible model for which we had a successful prediction was the histone chaperone CIA/ASF1-double bromodomain complex (PDB: 3AAD), with an RMSD between the known unbound and bound state of 4.37 Å. The 46 kDa complex formed by thioredoxin reductase (thioredoxin) and the NADP+ analogue AADP+ (PDB: 1F6M) was more flexible, exhibiting domain movements associated with an RMSD of 4.9 Å. As no top 10 model produced by JabberDock had an RMSD lower than 10 Å from the known bound state, this case was declared unsuccessful. However, a top 10 model featured an $f_{\text{nat.}}$ of 0.419 (rank 7, see Table S1), indicating that the binding site was partially identified. Thus, while in terms of RMSD several cases were unsuccessful, JabberDock could still identify their binding site, as shown in Figure 3.3.2(b).

3.3.2 Discussion

An isosurface derived from a STID map representation is suitable for the definition of an accurate protein-protein docking scoring function. Our results show that JabberDock can predict water-soluble complexes on par with a competitive range of blind protein-protein docking software and is highly robust across a range of difficult cases, an achievement not observed in other docking algorithms. The atomic models themselves are built using the last snapshot from the pool of conformations explored by the binding partners in their respective MD simulations. Combinations of other conformations within the monomers' simulations (see Figure 2.3.6) and dimeric arrangements within the full collection of candidate assemblies may be closer to that found in the crystallised bound state.

The strength of JabberDock to yield comparable results across the dataset indicates that the ability of the STID maps to encapsulate high-frequency atomic motions allows it to accommodate different levels of flexibility in interacting proteins. In complexes characterised by flat and relatively featureless binding sites, the surface complementarity function is likely to fail in highlighting a single most suitable docking position. On the other hand, when an interface is exceedingly complex, small perturbations about the docked pose are likely to lead to clashes, hindering the optimiser's exploration of this region of the energy landscape. These represent JabberDock's boundary conditions. The successful (and most typical) docking cases feature topographical complexities that enable both the scoring function and the optimiser to work harmoniously and effectively. This is the significant middle ground where the coupling of POW^{er} and STID isosurfaces provides excellent results, as indicated by the prediction of accurate protein complexes for most of the benchmark. These observations are expected to hold for any complex not requiring refolding or domain-level movements at the interface between binding partners.

3.3.3 Methods

3.3.3.1 Search space definition

Prior to docking, the two input monomers are centred at the origin *via* their centres of mass. With respect to the seven-dimensional search space, the x, y and z translation values are limited by the size of the receptor, the axis of rotation is constrained between values ranging from -1 to 1, and the rotation angle in radians ranges between 0 and 2π. The navigation of the PES and generation of models follows the rest of the procedure outlined in Section 3.2.1.1.

3.3.3.2 Dipole alignment re-ranking process

In a post-processing phase, the dipole map of the ligand is aligned to its atomistic counterpart in each of the 300 representative docked poses. An additional dipole score, S_{D}, is then calculated between the dipole map of the ligand and that of the receptor:

$$S_{\mathrm{D}} = \frac{\{D_{\mathrm{AB}}\} + \{D_{\mathrm{BA}}\}}{2}, \tag{3.3.1}$$

where the curly brackets refer to a median of a set of dipole score terms, with each individual dipole score given by:

$$D_{\mathrm{AB}} = \sum_{k=0}^{n} D_{\mathrm{A}} \cdot D_{k,\mathrm{B}}. \tag{3.3.2}$$

D_{A} is a specific dipole in A, $D_{k,\mathrm{B}}$ a dipole in B, n is the number of dipoles in B that are within 2 Å of D_{A}.

3.4 Transmembrane protein docking

Transmembrane proteins play an essential role as a mediator for many functions critical to an organism's survival. Situated within a lipid membrane that compartmentalises two distinct biological regimes; their tasks include sensing, signalling, motility, endocytosis and anchoring. Their malfunction is responsible for a multitude of diseases,[12] and consequently, they are a frequent target in drug design. The formation of complexes, wherein two or more transmembrane proteins will oligomerise into either helix bundles or β-barrels, is of vital significance to both the function and malfunction of these processes.

As we reviewed in Chapter 1, a plethora of increasingly sophisticated protein-protein docking approaches have been developed to address the problem of protein assembly prediction.[1] These efforts are nucleated around the community-led CAPRI competition, which is used to identify the most reliable algorithms, promising methodologies and current hurdles.[13] As discussed in Chapter 1, the majority of these methods centre around the docking of two or more water-soluble proteins, with only four methods engineered towards the task of docking transmembrane proteins, returning success rates inferior to that achieved by water-soluble docking methods. Furthermore, as largely proof of concepts, these approaches have given little consideration to the flexibility of the proteins. Given the challenges associated with the determination of transmembrane protein quaternary structure, it is clear that there is significant room for improvement in this niche field. As we have shown in Section 2.5, an isosurface derived from our STID map, at the ideal cut-off of 0.43 (see Section 2.3.4), is representative of a protein in its native environment. Therefore, the ability to generate STID maps simulated in any arbitrary environment makes them an attractive representation for membrane proteins, exposed to a biphasic environment. This fundamental characteristic, coupled with the success of JabberDock in Section 3.3, has highlighted JabberDock as an attractive solution to the lack of competitive software in the transmembrane docking arena. A STID map representative of a transmembrane protein can be obtained by independently simulating the pre-oriented partners immersed in an explicit lipid bilayer. Docking then requires maximising the complementarity of two membrane protein surfaces, with the ligand's translational motion perpendicular to the membrane and rotations into the plane of the bilayer constrained. Herein, we present and test our methodology to dock integral membrane protein dimers, made available in JabberDock as an automated pipeline.

3.4.1 Convergence of side-chain motion

There does not yet exist a standard transmembrane protein docking benchmark equivalent to the soluble proteins one made available by the CAPRI community.[6] To test JabberDock, we selected all the unbound cases involving pairs of transmembrane proteins within Memdock,[14] HADDOCK[15] and DOCK/PIPER[16] benchmarks. To avoid testing against similar examples, and thus biasing our statistics, we only selected one representative within test cases featuring >80% sequence homology. This resulted in a diverse benchmark set featuring 20 α-helical complexes of 2 easy cases, 15 medium, and 3 difficult cases under the CAPRI classification (see Section 3.2.3.1). Full details for each case in our benchmark set, including their sequence identity and three metrics used to define success by the CAPRI community (RMSD, $f_{\text{nat.}}$ and RMSD_I – see Section 3.2.3.2), are given in Table S2.

The shape of a STID map depends on the length of the Molecular Dynamics (MD) simulation used to build it. In Section 2.3.2, we compared the Pearson cross-correlation coefficient (CCC) of STID maps generated from increasingly long trajectories and observed that this metric converged after 600 ps. This timescale represents the time required for a protein's side-chains to explore their local conformational space in a water solvent. Since side-chain dynamics are slower in a membrane than in water, longer convergence times should be expected. Therefore, here we performed the same benchmark described in Section 2.3.2 for the membrane proteins in our benchmark set to ensure that the surfaces used for docking are fully representative of their biphasic environment. As in our previous work, we assumed convergence using the same arbitrarily high CCC value of 0.9997, reported between only a STID map's non-zero values.

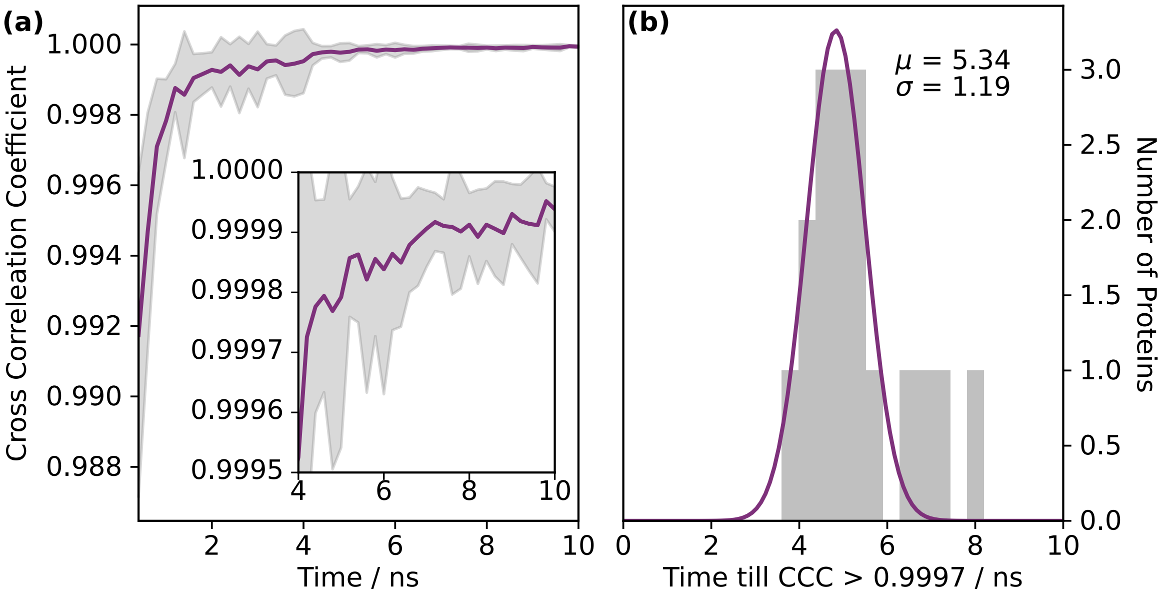

Figure 3.4.1: (a) Cross-Correlation Coefficient (CCC) between successive STID maps built from increasingly long timescales. The palatinate line indicates the average of 17 unbound distinct membrane proteins used for docking, the grey region represents the standard deviation at each trajectory length. The inset is a zoom into the top 4% of the figure. The CCC is calculated between non-zero values. **(b)** The number of proteins achieving a consistent CCC of 0.9997 by the listed time. A Gaussian (palatinate) has been fitted to the histogram. All protein motion converged within 9 ns, with the majority in the first 5 ns. The mean (μ) and standard deviation (σ) are reported.

Figure 3.4.1(a) illustrates the evolution of CCC of consecutive STID (*i.e.* generated with increasingly long MD simulations) for 17 unique proteins used for docking. The CCC increases rapidly over a few ns, before asymptoting towards one. The Gaussian fitted in Figure 3.4.1(b) shows that, on average, CCCs converge in ~5.5 ns, but some require upwards of 8 ns. This simulation time is long enough to account for all side-chain motions but short enough that no significantly large conformational changes can occur. Thus, we extract the last 9 ns of proteins simulated in a lipid bilayer to generate suitable STID maps.

3.4.2 Environment-independency of the STID isovalue

A key characteristic of our STID maps is that the ideal isovalue of 0.43 emerged naturally from the MD simulations in Section 2.3.4, specifically from the relationship between the surface accessible solvent area and the average STID value. This isovalue, as shown through the benchmark in Section 3.2.1.2, also independently serves as the optimal parameter to return the highest score at the known binding site *versus* other cut-offs. Since this value was derived for proteins in solution, we performed a similar benchmark here.

Figure 3.4.2 shows the variation in the surface complementarity score with cut-off in the known binding site, normalised and averaged across all 20 cases in our benchmark set. Again, we find a value of 0.43 to be most appropriate to return the highest quality results. Thus, crucially, the STID isovalue is environment-independent.

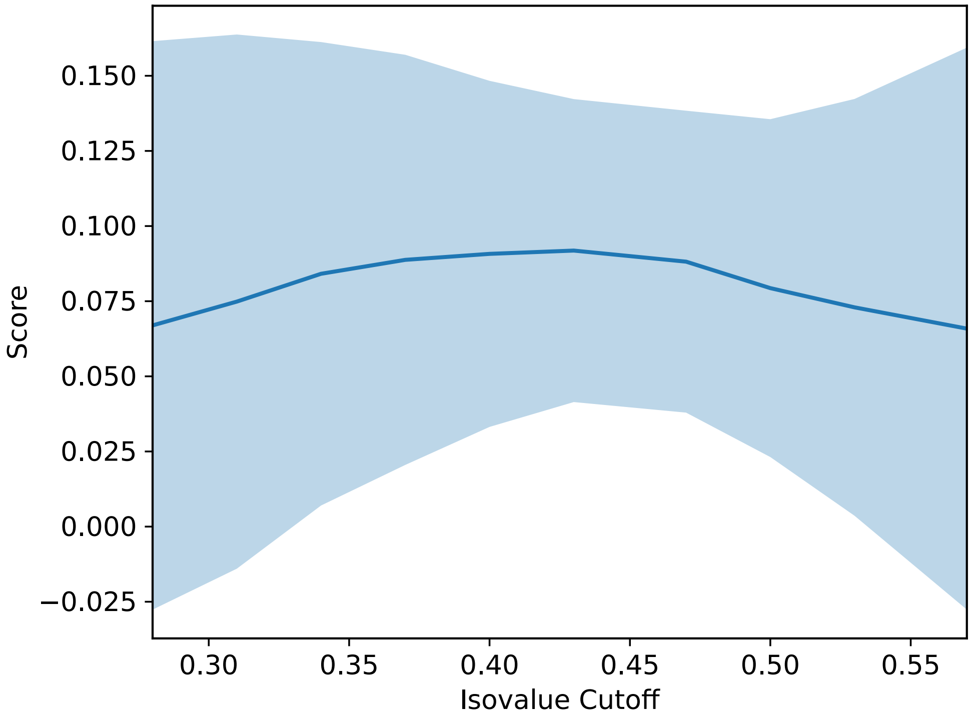

Figure 3.4.2: Mean variation in normalised surface complementarity score across the 20 protein cases with isovalue cutoff choice. The dark central line indicates the mean, the shaded region the standard deviation. The maximum (indicating what on average yields the best shape complementarity) occurs at a cutoff of 0.43.

3.4.3 Results

JabberDock docks transmembrane proteins *via* a multi-stage process summarised in Figure 3.4.3 and detailed in Section 3.4.5.1. In short, JabberDock requires input protein structures to be aligned with the centre of mass for the transmembrane region of the proteins at $z = 0$, where the z-axis is perpendicular to the bilayer plane. In our tests, we obtained these pre-orientated structures *via* the OPM server.[17] Structures are first automatically repaired, immersed in a POPE bilayer, and subjected to a short MD simulation enabling the generation of STID maps. The maps of both binding partners are then converted into isosurfaces and docked such that their surface complementarity is maximised (see Section 3.2.1.2). We explore the search space *via* a modified PSO (see Section 3.2.1.1), with the defined search space from the water-soluble benchmark altered to respect the constraints imposed by the lipid bilayer (see Section 3.4.5.2).

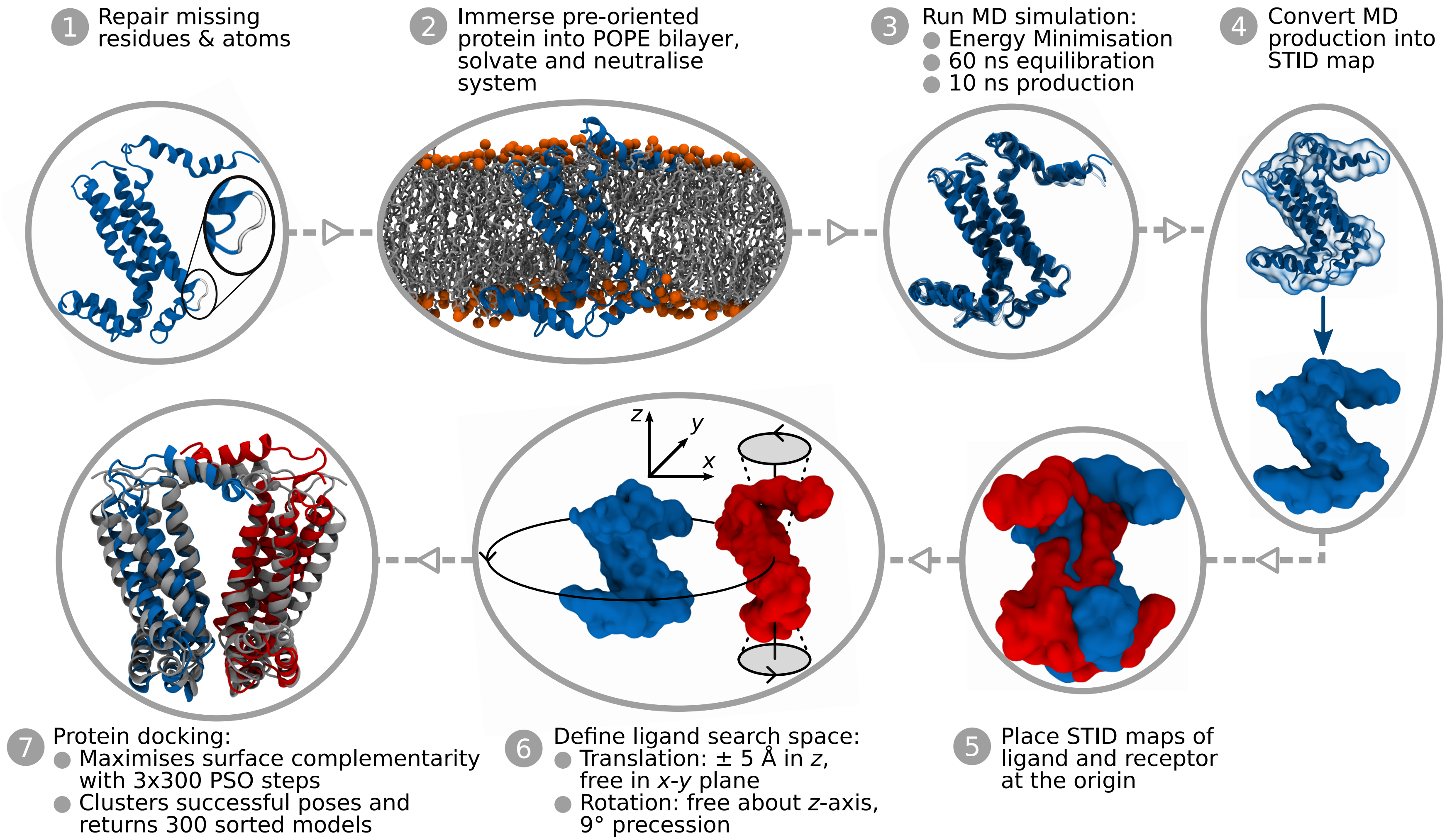
1 Repair missing residues & atoms
2 Immerse pre-oriented protein into POPE bilayer, solvate and neutralise system
3 Run MD simulation:
● Energy Minimisation
● 60 ns equilibration
● 10 ns production
4 Convert MD production into STID map
5 Place STID maps of ligand and receptor at the origin
6 Define ligand search space:
● Translation: ± 5 Å in z, free in x-y plane
● Rotation: free about z-axis, 9° precession
z
y
x
7 Protein docking:
● Maximises surface complementarity with 3x300 PSO steps
● Clusters successful poses and returns 300 sorted models

Figure 3.4.3: JabberDock transmembrane protein docking pipeline. Full details of each step are available in Section 3.4.5.1. This example's target complex is the homodimer 1Q90(BF), using 2ZT9(A) as the receptor (blue) and ligand (red). Step 7 features a representation of the 5th best model; an intermediate success overlaid on the bound structure (grey).

To first demonstrate the utility of our surface-based scoring function over atomistic energy profiles, for all cases in our benchmark set we generated 100 new poses by applying small, random perturbations (< 2.5 Å) in ligand positions from their known bound state. We then evaluated their STID map-based score, as well as their atomistic and coarse-grained van der Waals energies. The coarse-grained vdW score was calculated from the backbone atoms, with typical MARTINI parameters[18] used for the forcefield. We finally compared all these scores with their value at the known bound state, reporting them in terms of a percentage increase or decrease. The results of this comparison are shown for three examples in Figure 3.4.4. The STID score provided distinguishable gradients to the binding site, with a maximum at the ground truth. In contrast, the atomistic vdW energy has no discernable trends, with positions away from the binding site having arbitrarily higher or lower energy. Furthermore, these energies featured extremely steep gradients, requiring the use of a log-scale for a manageable comparison. The coarse-grained vdW score features a smooth gradient to the binding site for the formate channel case (PDB: 3KCU), but returns a similar result to the atomistic vdW score, albeit significantly less steep, for the other two. van der Waals scores for all proteins in our dataset behave consistently with the examples shown here. Our STID score behaves as exemplified for 15 out of 20 cases. In the remaining five cases a gradient is not clearly distinguishable, although percentage changes always remain in the same order of magnitude as demonstrated here. Thus, our surface-based STID scoring function is effective as it bypasses the need to explicitly handle packing of interfacial atoms, yielding smoother and gentler gradients compared to typical atomistic representations (see Figure 3.4.4).

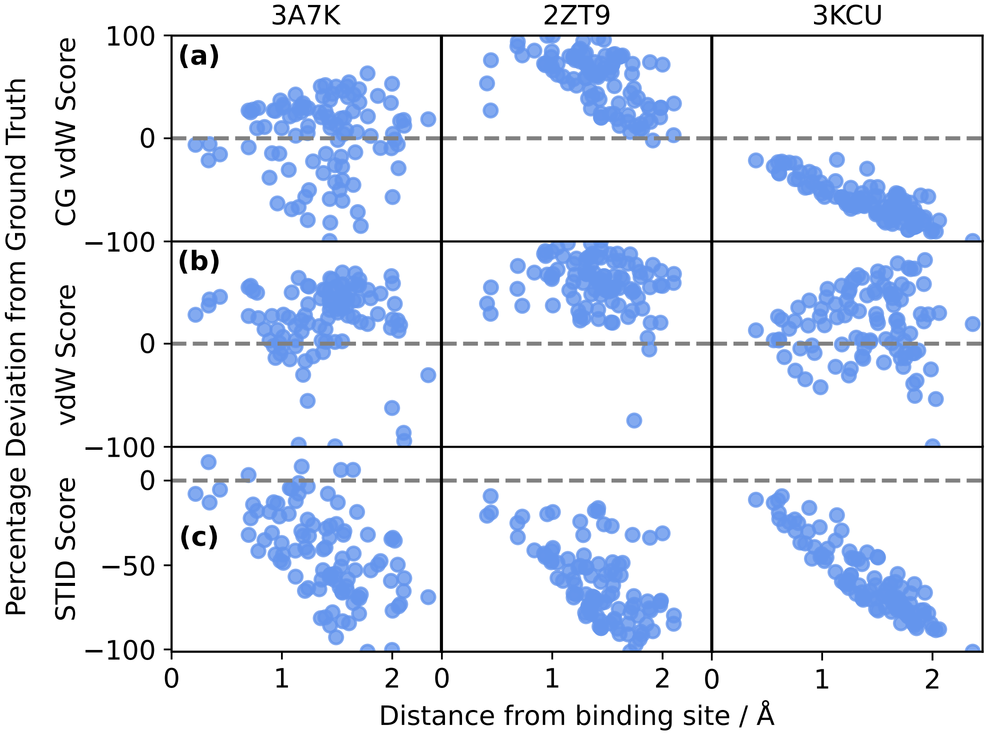

Figure 3.4.4: Comparison between atomistic docking scores, represented by **(a)** coarse-grain, **(b)** all-atom van der Waals energy, and **(c)** our STID surface complementarity-based score. For three different transmembrane protein complexes, we perturb the position of the known docked pose, reporting the percentage deviation of scores from their value at the bound state. For clarity, percentage for atomistic van der Waals energies are plotted in a log-scale. Our STID score produces clearer and distinctly gentler gradients.

Challenged with the docking benchmark, JabberDock was successful (*i.e.* yielding at least one acceptable model or better among its top 10 candidates) in 75.0% of cases in our benchmark set, producing an intermediate quality success in 40% of cases (see Figure 3.4.5, Figure 3.4.6, and Table S2). This remarkable performance compares favourably against other transmembrane protein docking software and is an improvement over JabberDock's performance against water-soluble proteins (54 %). This result is explained by JabberDock's ability to identify the binding interface correctly, primarily due to its sensitivity to the dynamics of individual amino acids. As shown in Figure 3.4.6(a), in nearly every test case, at least one prediction in the top 10 results features a correctly identified binding interface. Given that our STID map-based scoring function performs comparatively in a water and membrane environment (see Figure 3.4.2), this substantial increase in the success rate can be explained by the added benefit of *a priori* knowledge about the orientation of the proteins with respect to the bilayer, coupled with the strict constraints imposed by the lipid membrane.

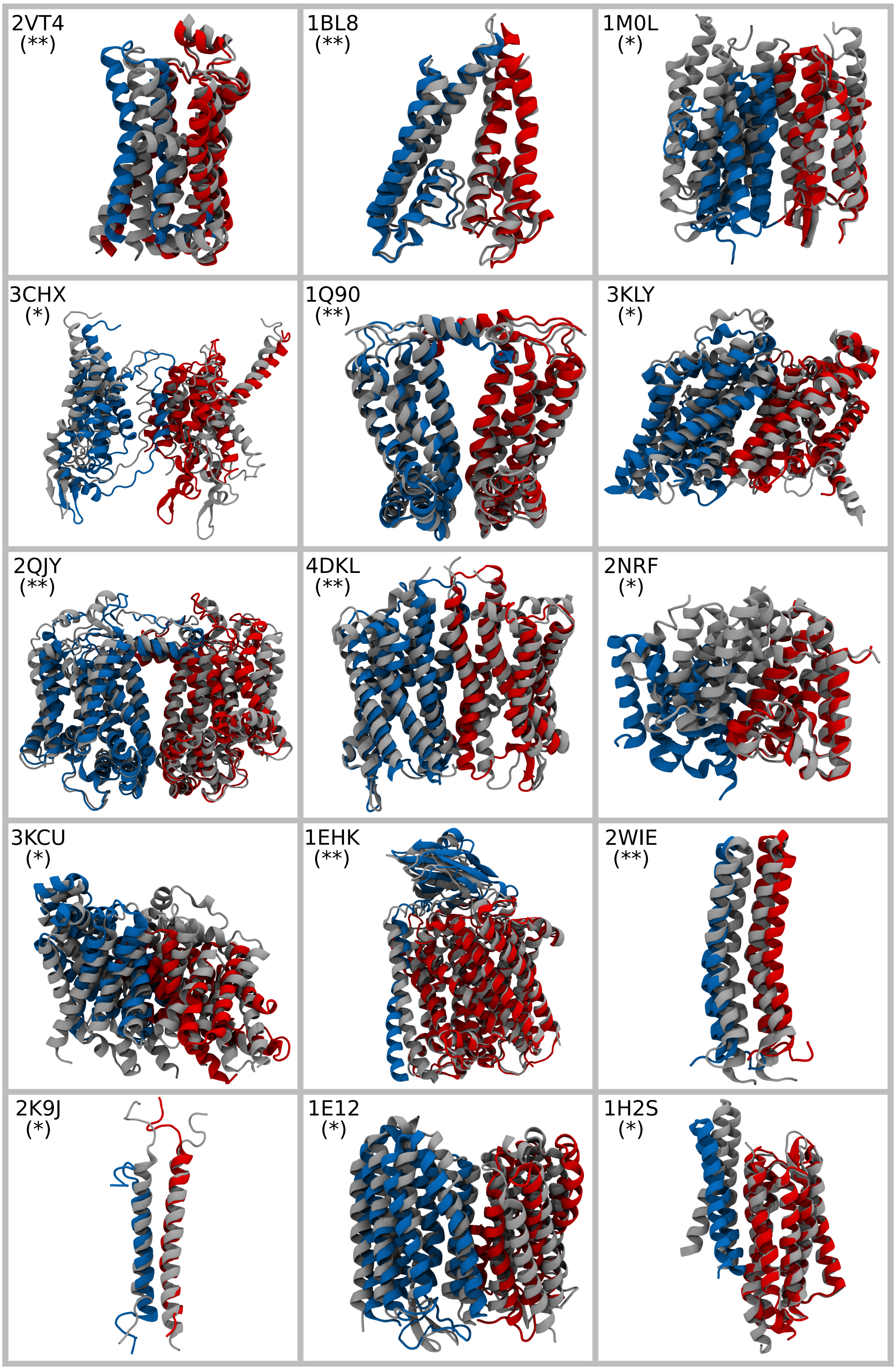

Figure 3.4.5: Best model in the top 10 for every successful case (noted as * if acceptable, and ** if intermediate). The known bound state (indicated by its PDB code) is shown in grey, the receptor and ligand used for docking in red and blue respectively. See Table S2 for details.

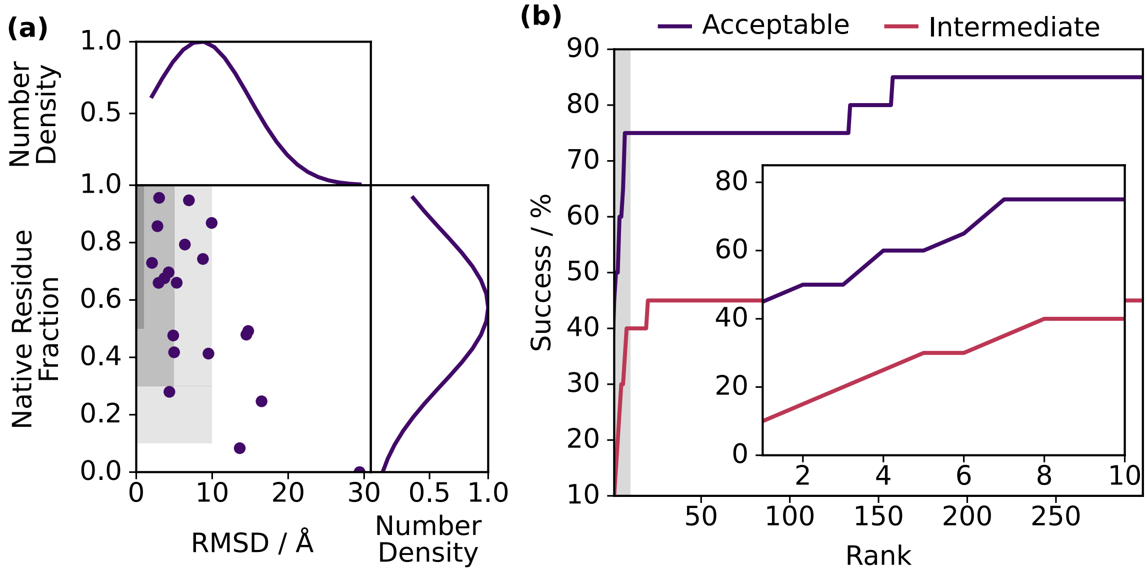

Figure 3.4.6: **(a)** Quality of best models within the top 10 results for every docking case. For each case, the lowest α-carbon RMSD between the prediction and crystallised homolog is presented against the associated native residue fraction ($f_{\text{nat.}}$). The dark- to light-shaded regions represent the criteria for high to acceptable quality results. **(b)** Percentage of cases yielding an acceptable (blue) and intermediate (pink) success as a function of the number of ranked structures considered as candidate models. The region corresponding to the top 10 models is shaded and magnified in the inset.

Expanding the pool of candidate structures to the whole 300 models returned by JabberDock does little to improve its overall success rate (85.0%, see Figure 3.4.6(b)), in contrast to other protein docking software and our soluble benchmark. This is because JabberDock returned a top 10 successful model for the majority of cases (15 out of 20). The few unsuccessful cases, also challenging for other docking algorithms, possess similar structural features to those complexes in the water-soluble benchmark that JabberDock found problematic. For example, the NavAb voltage-gated sodium channel (PDB: 3RVY) features an interlocked arrangement where, following the unbound MD simulation, the binding site closed up, preventing the ligand from navigating into the binding pocket.

We find that the majority of other unsuccessful cases, both in the water-soluble and transmembrane protein datasets, have binding interfaces devoid of “feature-rich” regions on the STID isosurfaces. By this, we mean that the surface is relatively flat, with an absence of “bumpiness”. The ability of JabberDock to consider flat surfaces is further hindered when a docked pose features some minor shape intersection, which nullifies subtle detail on the surface. The work in Section 4.4 is focussed on improving the scoring function to better consider the topography of the isosurface at this scale. We illustrate two different surfaces; one with a flat surface, that of wild type cytochrome c oxidase (PDB: 1M56), and one with a bumpier surface, that of cytochrome b6f (PDB: 1Q90) in Figure 3.4.7. This bumpiness is caused by both the flexibility of the backbone, and the diversity of amino acids present at the binding site, with a variation of those with short and long side-chain resulting in more topographically distinct surfaces.

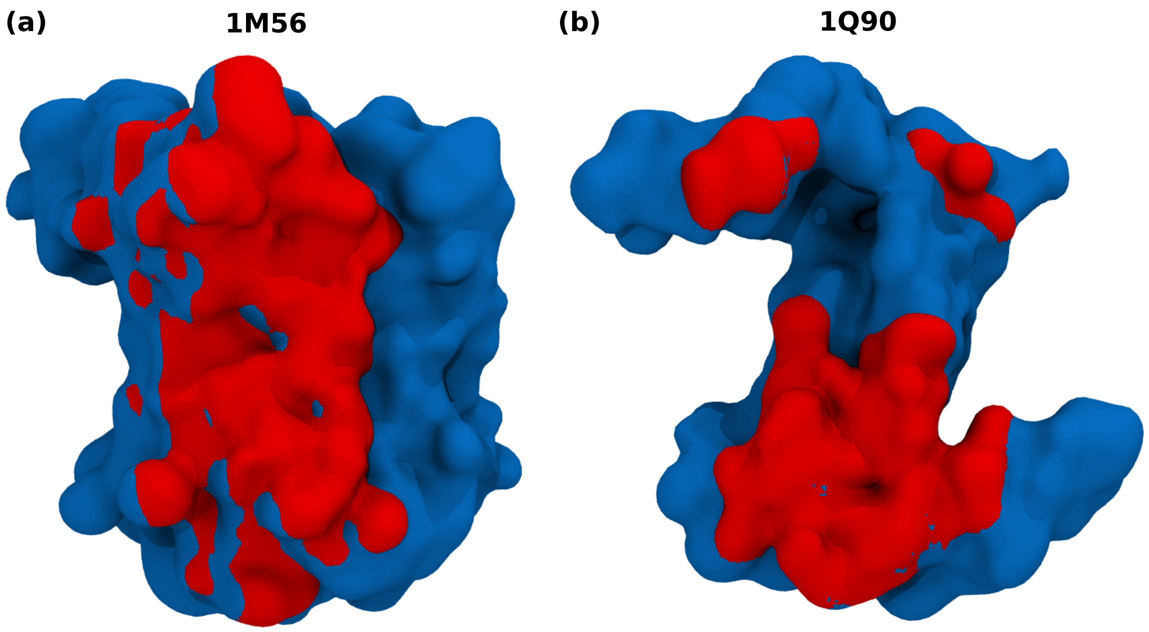

Figure 3.4.7: STID map isosurface at 0.43 cutoff for two cases, with their binding site indicated in red. **(a)** Wild type cytochrome c oxidase (PDB: 1M56) receptor, with B = 0.769(1). JabberDock was unable to identify a single acceptable model for this case. **(b)** Cytochrome b6f (PDB:1Q90) receptor, with B = 0.724(2). JabberDock identified an intermediate quality result at rank 5 for this complex.

Since the level of bumpiness can be considered subjective, we attempted to quantify it by measuring the similarity of local regional surfaces to a plane. Specifically, we first isolated the binding site STID map from the rest of the protein (indicated in red in Figure 3.4.7), which we define as residues within 5 Å of the binding partner, for all proteins in our dataset. We then tessellated the surface at our optimised cut-off of 0.43 with a series of triangles using a marching cubes Lewiner algorithm. Each triangle has a surface area of 0.5 Å^2. For the vector normal to each triangle, we calculate the mean dot product with all other neighbouring triangle normals over a 10 Å^2 surface area as a measure of the local bumpiness value (B). The mean of all localised B gives the overall bumpiness of the binding site surface. A value closer to 1 indicates that the surface is more planar, while 0 is more spherical. Table 3.4.1 provides the corresponding B for each test case. We find that the majority of proteins involved in cases that succeeded *(i.e.* a successful model in the top 10) return a lower B value overall ($B = 0.74(2)$, averaged across all successful models), while cases which were not so successful feature more planar surfaces ($B = 0.78(1)$). While the difference is small, it is statistically significant.

Table 3.4.1: Bumpiness of each protein (both receptor and ligand if different), with the chain indicated in the parentheses and the rank of the first successful model (if applicable). The average for successful test cases is $B = 0.74(2)$, while that of unsuccessful models is $B = 0.78(1)$.

PDB Code	Bumpiness, B	First Successful Model
1K4D (C)	0.777(2)	2
3S33 (A)	0.752(1)	1
3S33 (B)	0.741(1)	1
1GU8 (A)	0.737(2)	1
2F95 (B)	0.751(2)	1
3V3C (A)	0.807(1)	1
3A7K (A)	0.742(2)	2
3OMI (A)	0.769(1)	X
1QLE (C)	0.771(1)	X
2ZT9 (A)	0.724(2)	5
1YQ3 (C)	0.765(2)	159
1YQ3 (D)	0.770(3)	159
1ZRT (C)	0.743(2)	3
1YEW (B)	0.739(2)	8
3KCU (A)	0.746(2)	5
3ODU (A)	0.805(2)	X
3RW0 (A)	0.759(2)	X
4EA3 (A)	0.755(1)	1
2IC8 (A)	0.736(2)	1
2Y00 (A)	0.759(2)	8
2Y00 (B)	0.756(1)	8
3Q7K (A)	0.775(1)	1
1C8S (A)	0.715(2)	7
2RMZ (A)	0.714(1)	1
2K1A (A)	0.722(3)	1
2N2A (A)	0.768(4)	135
2M0B (A)	0.785(2)	135

There are exceptions to this flatness rule, including that of 3RVY. Thus, while the bumpiness of a STID isosurface can play a role in the success of an individual case, it is not necessarily the sole reason behind unsuccessful cases, hence the similarity of the two average bumpiness values reported above for successful and unsuccessful cases. Besides occlusion of the binding site or a flat surface, either as a combination or individually; remaining unsuccessful cases featured relatively small binding interfaces, which are particularly demanding to identify given the goal of the optimiser to maximise surface complementarity.

3.4.4 Discussion

We have presented a pipeline enabling our blind water-soluble protein-protein docking software, JabberDock, to successfully tackle cases involving integral membrane protein dimers. This success is due to the molecular representation we adopt to dock proteins, STID maps; casting electrostatics, dynamics and protein's shape into a single volumetric representation. The preliminary stages in the building of a STID map require an MD simulation; thus, the different characteristics expressed by the protein in both the soluble and lipid environments are encapsulated in the isosurface's topography. Consequently, other than an extended MD simulation, one only needs to restrict the search space of the ligand in the docking protocol to regions occupied by the lipid membrane. The problem is, therefore, more manageable than a water-soluble protein docking one.

As no standard transmembrane protein docking benchmark exists, we applied JabberDock to an unbound benchmark of 20 transmembrane α-helical proteins taken from three other benchmarks,[14–16] which returned a success rate of 75.0%. This is a significant improvement over the 54% found for the water-soluble protein docking benchmark. Improvement here, therefore, is primarily due to the reduced size of the search space. In contrast to the water-soluble docking pipeline, the atomic models returned are built from the original crystal structure, as we found that returned higher-quality predictions overall. Our results correspond to correctly identifying 7 *versus* DOCK/PIPER's 2 out of 8 cases,[16] 8 *versus* Memdock's 4 / 11 cases,[14] and 1 *versus* HADDOCK's 1 / 3 cases[15] (note that two cases were tested by more than one of these methods, hence 22 individual comparisons from 20 cases). Applying the same difficulty classification method employed by CAPRI to their benchmark[6] (see Section 3.2.3.1), we see that acceptable models within the top 10 candidates were obtained even for some of the most flexible cases. Unsuccessful cases were primarily those where the STID maps featured flat interfaces, a similar issue encountered with the water-soluble benchmark set. The boundary conditions imposed by this "flatness", was the motivation behind the dipole-alignment re-ranking method in Section 3.3.1, and is the focus of much of the work in Section 4.4. In addition, the work in Section 4.4 also focuses on addressing the other major boundary condition introduced through the optimiser: navigating into occluded binding sites.

Future work to further demonstrate the applicability of JabberDock to transmembrane protein docking will need to address the formation of β-barrels and lipid plugs. Lipid plugs, particularly those with a significant quantity of lipids modulating the binding interface,[19] pose a particular challenge, as the absence of lipid-atoms in the STID maps, will result in poor surface contact at the known binding pose. Including lipid-atoms in the STID map is possible; however, this would require prior knowledge of the binding interface position to avoid losing molecular detail elsewhere. Regardless of these challenging regions of the proteome, given the success of the results presented here and that previously demonstrated with globular proteins, we expect JabberDock to also perform well with transmembrane-solvent proteins, regardless of whether the ligand is extracellular, periplasmic or cytoplasmic.

3.4.5 Methods

3.4.5.1 Generation of transmembrane protein surfaces

Since STID maps are generated from MD simulations of proteins in their native environment, we need to first explicitly represent a membrane to simulate the transmembrane protein in. Here, we detail the operations required to prepare the binding partner STID maps, corresponding to steps 1, 2 and 3 of Figure 3.4.3. Proteins must be pre-oriented before input, *i.e.* the centre of the transmembrane domain of both binding partners is at the origin with the appropriate orientation given that the bilayer will be built parallel to the $x - y$ plane. Such pre-alignment comes as standard for structures downloaded from the OPM server.[17]

1. Structures are (optionally) initially checked and, where necessary, repaired using Modeller.[20] Specifically, the FASTA sequence of the protein is downloaded from the PDB database[21] (placing the FASTA file in the folder is enough if there is no connection to the Internet), and used to patch up to 15 consecutive missing residues.

2. The protein is immersed in a POPE bilayer and solvated *via* the PACKMOL-memgen tool[22] available through the AmberTools(v.18+) package. Lipid and TIP3P water molecules are placed using a random seed, and 80 loops are performed during PACKMOL's GENCAN routine to improve packing with a total of 120 NLOOPS for all-together packing. A tolerance of 2.4 Å is used to detect clashes between molecules. POPE residue names are then corrected to reflect the SLipid[23] nomenclature before the topology files are generated through GROMACS.[7] A

small fix is used to accommodate the different angle and dihedral descriptions between SLipid and Amber ff14SB.[24] Finally, the system is neutralised with Na^+ or Cl^- counterions.

3. Membrane protein systems are simulated using GROMACS[7] MD engine, with Amber ff14SB[24] and SLipid[23] force fields describing the protein and lipids respectively. The system is energy minimised *via* steepest descent within a tolerance threshold of 200 kJ mol^{-1} nm^{-1}, with the maximum number of steps set to 5×10^6. Cut-offs for both Coulombic and van der Waals interactions are set to 1.2 nm, with particle mesh Ewald summation treating long-range interactions. The system is then equilibrated for 20 ns within an isothermal-isobaric ensemble, with a 2 fs timestep and all bonds constrained with the LINCS algorithm.[25] A semi-isotropic Berendsen barostat is used to maintain a pressure of 1 bar, with a coupling constant of 1.0 ps and compressibility of 4.5×10^{-5} bar^{-1}. The temperature is set to 310.15 K. A further 50 ns is then simulated, with the same settings but with all constraints removed. Atomic coordinates of the last 10 ns (for reasons shown in Figure 3.4.1) are saved every 5 ps, and used to generate a STID map following the procedure described in Section 2.2.

3.4.5.2 Search space definition

Here we provide details on the docking process of protein surfaces generated from STID maps, corresponding to steps 5, 6 and 7 of Figure 3.4.3, where constraints are imposed on the search space discussed for the water-soluble proteins in Section 3.3.3.1 due to the presence of the lipid membrane. An initial starting point with the centres of mass of the two input monomer transmembrane regions are fixed at the origin prior to generating any models. In the 7-dimensional search space, the x and y translation values are limited by the size of the receptor, and the ligand is only allowed to move ±5 Å along the z-axis. The axis of rotation is aligned along the z-axis, but is also permitted to precess by up to 0.157 radians (9°) into the xy-plane. Possible rotation angles in radians range between 0 and 2π. The navigation of the PES and generation of models follows the rest of the procedure outlined in Section 3.2.1.1.

3.4.5.3 Homology modelling

Several test cases only had their ligand and/or receptor starting structure known from a homolog, sometimes bound to an alternative binding partner. For these cases, receptor and ligand crystal structures were mutated into their target counterparts *via* the Modeller program.[20] Motifs up to 15 residues long were permitted to be patched if they were missing from the structure, and structures were kept frozen to prevent optimisation of models. The roto-translations returned by JabberDock were applied to these structures to yield the final predicted complexes. Table S2 reports on the sequence identity between homologs and the target structure. Their RMSD, determining case difficulty (see Section 3.2.3.2), is also provided. We note that three benchmark cases (1ZOY, 2VT4 and 1EHK) feature binding partners extracted from a known complex that is a homolog to the target. Although not a real-world test case, these are suitable benchmark cases as the conformations of subunits in the two dimers differ. The RMSDs reported in Table S2 refer to those between target structures and crystal structure, either of the unbound molecule or mutated structure from the homolog.

3.5 Applications

3.5.1 Transthyretin fibril formation

Transthyretin (TTR) is a homotetramer (see Figure 3.5.1) transport protein involved in various cardiac and neurological genetic diseases. This is due to the ability of misfolded TTR to form amyloid fibers.[26] The pathogenetic variants have shown to cause hereditary diseases such as Familial Amyloidoic Polyneuropathy and Familial Cardiomyopathy.[27–29] Figure 3.5.1 demonstrates how the common mutations associated with these diseases occur around or in the amyloid binding sites of TTR. The wild-type has also been shown to generate amyloids leading to Senile Systemic Amylodosis.[30] Designing an algorithm capable of auto-tiling a fibril such as TTR in 3D, in the pursuit of developing medicines to prevent amyloid aggregation, was the subject of a project by a visiting PhD student from Université de Lyon, Lorenza Pacini, in July of 2019.

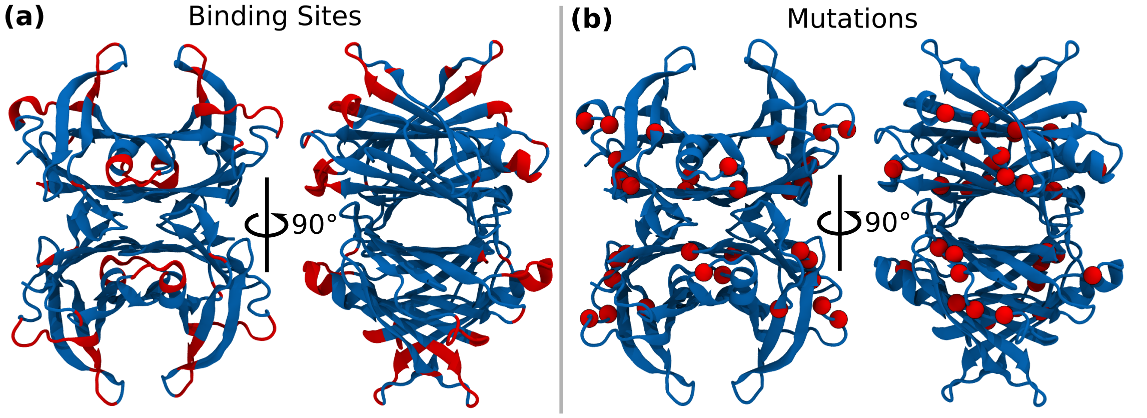

Figure 3.5.1: Transthyretin homotrimer (PDB: 1F41). **(a)** Binding site location associated with TTR amyloid aggregation.[31] **(b)** Common single-point mutations known to catalyse amyloid aggregation.[31,32]

The novel auto-tiling method defines several fibril types depending on the oligomeric interaction site of TTR and the topology of the final arrangement (see Figure 3.5.2 for predicted examples). Tiling the homotetramers into these fibril types requires prior knowledge of the general repeating unit, and thus an initial roto-translation to map the homotetramer into the candidate binding position. Once the fibril is built, it is then possible to compare with electron microscopy (EM) negative stain images to confirm the validity of the model.[33] TTR can form multiple structurally distinct fibril networks,[31] with the different types of fibres also depending on whether the TTR is wild-type or not. There are, therefore, three key concerns that must be addressed before the auto-tiling can take place:

1. Initial docked candidates are required with associated roto-translations.
2. A diverse pool of poses are needed to meet the diverse fibre model requirements.

3. These predicted arrangments need to consider possible mutagenesis.

JabberDock can deliver on all of these fronts. It has shown to provide accurate docked predictions for soluble proteins (Section 3.3), and the K-means geometric clustering yields a distinct set of solutions. We have already discussed how STID maps reflect the changing dynamics of individual amino acids following some distal binding (see Section 2.6), it would be expected that this property would also serve to measure the impact of possible mutations. A topographically distinct STID map would accordingly affect the docking and, thus, the candidate fibril models produced. Therefore, we applied JabberDock to the TTR wild-type (PDB: 1F41), before the auto-tiling procedure was applied to generate the different fibres. The results of some proposed models are given in Figure 3.5.2.

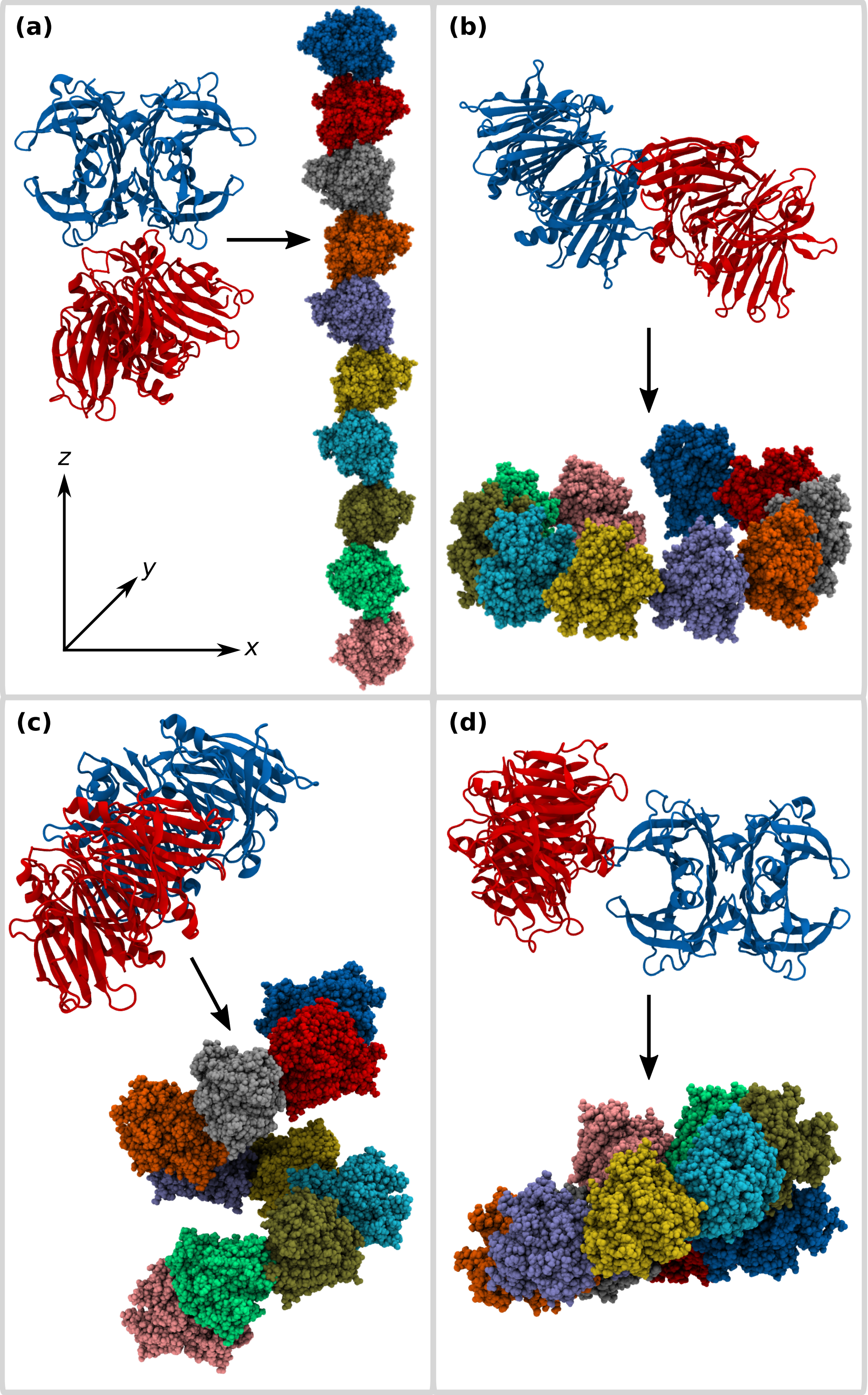

Figure 3.5.2: Four fibril models of TTR predicted through an initial starting docked pose provided by JabberDock. **(a)** Rank 4 docked pose, fibril Type 1. **(b)** Rank 1 pose, Type 2a. **(c)** Rank 12 pose, Type 2b. **(d)** Rank 94 pose, Type 2c.

The fibril types in Figure 3.5.2 are classified with the following:

1. Periodic stacking in z with minimal displacement into the $x - y$ plane, leading to a curvature of zero, Figure 3.5.2(a).
2. Stacking occurs with both a non-zero azimuthal and polar angle, leading to a fibril with a finite radius/curvature with respect to the translational z-axis, *i.e.* a helical fibril.
    (a) The helix pitch is smaller than the size of the TTR homotrimer along z, leading to a fibril of finite size that cannot grow due to steric clashes, Figure 3.5.2(b).

    (b) The pitch of the helix is larger than the TTR, leading to a helix, Figure 3.5.2(c)

    (c) The pitch of the helix and TTR size are similar, leading to a closed helical fibril, Figure 3.5.2(d)

In generating these fibres, the top 100 poses were selected for further analysis. The four fibers shown are those with interaction sites that corroborate experimentally proposed residue contacts[31] (see Figure 3.5.1). It is a testament to JabberDock's success that two of the top 10 models, including the highest-ranking, are in this set of four. Considering a greater pool of models enables a richer dataset and a greater number of plausible fibril models. Within these 100 poses are arrangements that provide variants on the types shown in Figure 3.5.2 that are equally feasible. Thus, utilising JabberDock as a preliminary step in the auto-tiling process allows for the rapid generation of distinct models that can then be compared with experimental data such as EM negative stain imaging, resulting in a highly detailed, atomistic representation of the fibre network not available elsewhere. Future work will look into mutant variants of TTR and other fibres.

3.5.2 Confirming Mass Photometry data for *bo*$_3$ oxidase dimer formation

Author's note: *the experimental results presented in this Section (Figure 3.5.3), were collected by the co-authors: Olerinyova et al. (see Publications & Manuscripts).*

Integral membrane proteins make up 20 to 30% of the proteome and play a significant role in several tasks such as nutrient transport through the membrane and signalling.[34] Despite this, they make up only 4% of structures in the PDB database.[21] This is due to the difficulties associated with studying and characterising them as they are unstable under typical aqueous conditions. Experimental strategies to investigate their properties and behaviour rely on retaining a native lipid-like environment. A diverse range of membrane mimetic systems are used to achieve this, such as nanodiscs, detergent micelles and amphiphols.[35] Since these systems are inherently heterogeneous with a multitude of interacting components, it is challenging to resolve the structure and function of an integral membrane protein, particularly at a single molecular level.

The recently developed Mass Photometry technique[36] has demonstrated its remarkable capability as a protein characterisation tool in competition with widely used techniques such as mass spectrometry. It has shown to accurately determine molecular masses, detect ligands bound to soluble proteins without a label, and resolve different oligomeric states. Mass Photometry works by measuring the interference between light reflected off an interface and light scattered by molecules attached to said interface, with the amount of interference related linearly to mass. This technique is non-invasive, requires minimal sample concentrations (<<µm) and volume (µl), it can report on populations due to its single-molecule nature and does not rely on indirect absorption measurements. Mass Photometry is, therefore, an attractive tool to apply to transmembrane protein structure characterisation.

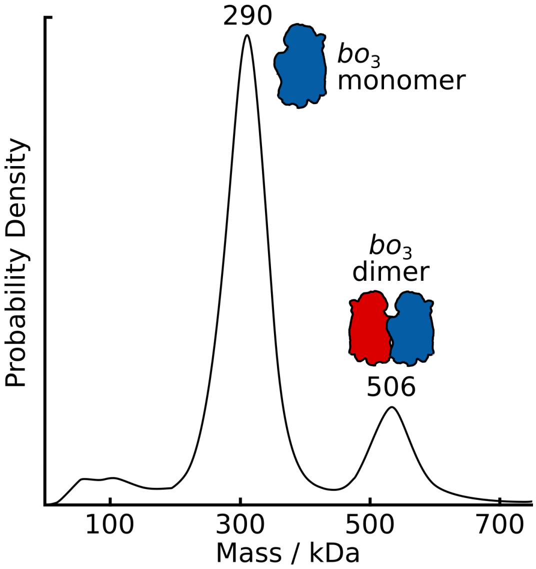

Figure 3.5.3: Mass Photometry detection of *E. coli* bo_3 oxidase monomers and dimers isolated from LMNG detergent micelles by dilution.

We demonstrated that Mass Photometry can be used in transmembrane protein structure prediction by corroborating Mass Photometry predictions with our own docking approach. In this context, the Kukura group at the University of Oxford gathered Mass Photometry data of *E. coli* bo_3 oxidase in lauryl maltose-neopentyl glyco (LMNG) stabilising detergent. The results indicated both a bo_3 monomer and a dimer. The dimer's presence is debated, with some results suggesting its existence,[37] while others indicate its absence.[38] The monomer and dimer were found at 80% and 20% abundance respectively (see Figure 3.5.3) by the Kukura group. The measured monomer mass of 290 kDa suggests the presence of 146 kDa of LMNG, equating to 146 individual molecules. Similarly, the measured dimer mass of 506 kDa would correspond to a 288 kDa dimeric protein with 218 LMNG bound molecules. Suspecting the dimer an artefact of the drop dilution method, analysis was also conducted at a higher concentration of LMNG above the critical micelle concentration. The results of this agreed with the original findings.

To test the feasibility of dimer occurrence, we generated potential bo_3 (PDB: 1FFT) dimer structures using the transmembrane docking routine of JabberDock. For this, we followed the procedure outlined in Section 3.4.5.1, by first simulating the monomer (PDB: 1FFT) in a POPE bilayer for 60 ns, and used this equilibrated structure to generate a pool of 300 possible dimeric arrangements (the best candidate is shown in Figure 3.5.4(b)).

We used these models to validate the Mass Photometry data by investigating the loss of detergent that would be required following the formation the dimer. Specifically, we calculated the transmembrane solvent-accessible surface area (SASA), assuming a transmembrane width of 30 Å as indicated by the OPM database,[17] *via* the Shrake-

Rupley ("rolling ball") algorithm.[39] By first calculating the SASA of the monomer's transmembrane region, we can derive a SASA per kDa occupied by the detergent. Assuming this quantity is invariant between monomer and dimer, we can infer the total mass of detergent bound to the dimer by measuring its SASA. The drawback of this method is that its accuracy hinges on an accurate prediction of the location of the water/POPE interface (inferred here using the OPM database). The transmembrane SASA of the monomeric bo_3 was found to be 21800 Å^2. Given the mass of LMNG, determined by Mass Photometry (146 kDa), we can calculate the ratio of SASA per kDa of detergent:

$$\frac{21800\ \text{Å}^2}{146\ \text{kDa}} = 149.4\ \text{Å}^2\ \text{kDa}^{-1}.$$

We then calculated the SASA for all dimer complexes generated by JabberDock, the distribution of which is shown in Figure 3.5.4(a), giving an average surface area of 33100 Å^2. Taking this as a consensus value, and assuming the ratio of bound lipids is consistent between monomeric and dimeric bo_3, we can derive the mass of LMNG that are able to pack around the dimer:

$$\frac{33100\ \text{Å}^2}{149.4\ \text{Å}^2\ \text{kDa}^{-1}} = 221 \pm 3\ \text{kDa}.$$

This prediction is remarkably close to the experimental value of 218 kDa. These data support the presence of the bo_3 oxidase dimer *in vitro* (see Figure 3.5.4(b)), and illustrate the accuracy of Mass Photometry in quantifying not only the mass of the polypeptide, but also the detergent mass bound to solubilised membrane proteins.

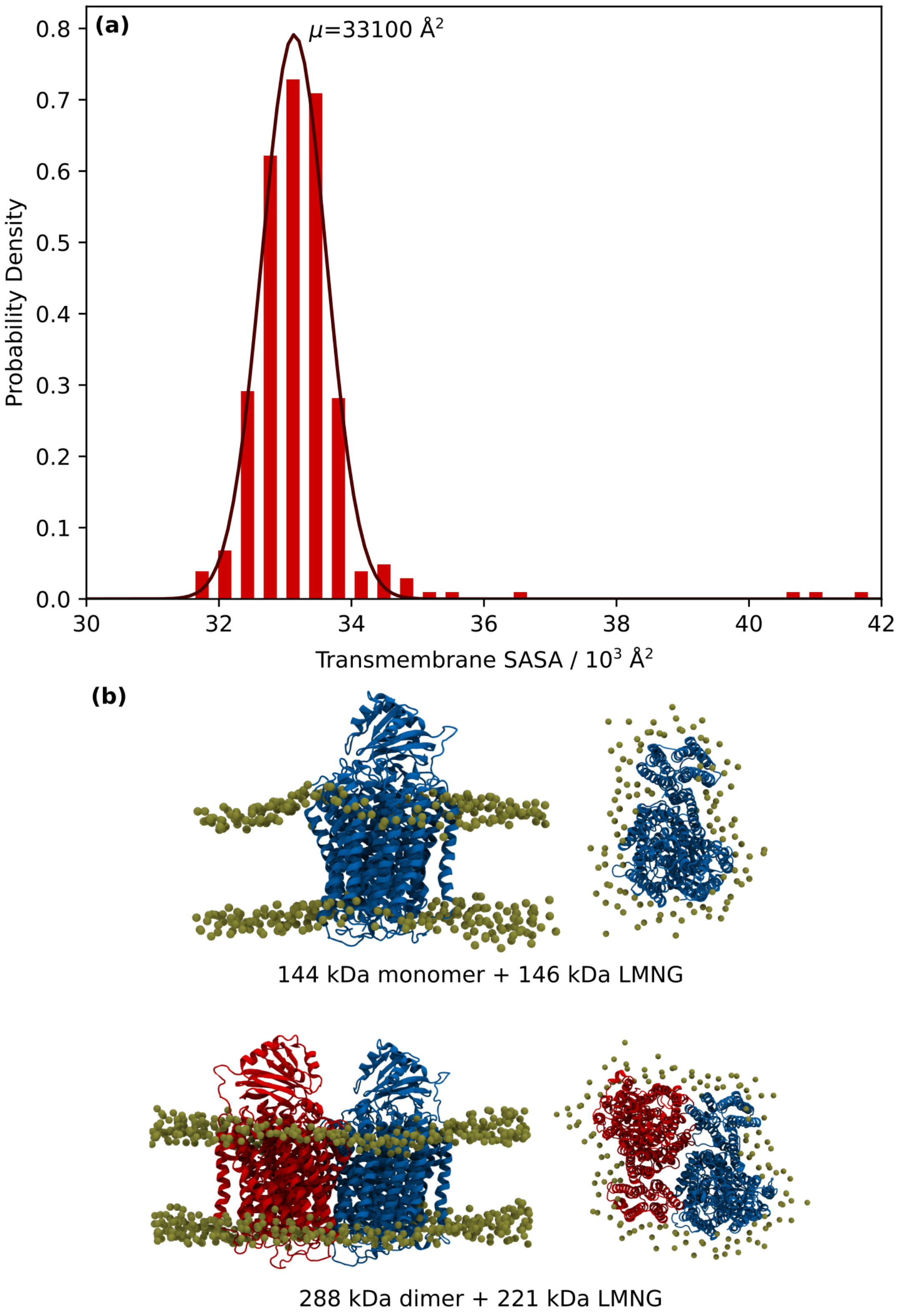

Figure 3.5.4: **(a)** Distribution of transmembrane surface area across all dimers predicted by JabberDock. The number at the peak indicates the mean of the data. Both a histogram (red) and a fitted gaussian (dark red) of the data are shown. **(b)** Monomer (top) unit of *E. coli bo*$_3$ oxidase and the corresponding best ranked candidate dimer (bottom).

3.6 Conclusions

Protein-protein docking is an immensely useful tool in biomolecular modelling as it provides oligomeric models not obtainable through traditional, *in vitro* means. Its inherent complexity has lead to the development of a wide-range of approaches over the last 25 years. However, despite the persistent advent of novel software, a global solution continues to elude the community. Adequately accomodating the flexibility of a protein, both on a domain and side-chain scale, is an issue that all protein-protein docking methods must account for. While domain-level shifts only occur for protein complexes that undergo an allosteric change, side-chain flexibility pervades all docking scenarios. Typically, atomic positions are treated as spatially fixed, and a local refinement takes places following rigid-body docking to accommodate this flexibility. These approaches, therefore, fail to consider the effect of side-chain flexibility during the actual docking procedure. In Chapter 2 we presented STID maps, a strategy to represent how a protein is perceived by its immediate surroundings. Our physical formalism encompasses the localised electrostatic nature of the space occupied by a molecule, and the dynamics of the protein itself, into a series of local dipole vectors, which is ultimately cast into a volumetric representation. Our STID maps are, therefore, an attractive solution to the protein-protein docking problem. STID maps can be constructed from a protein in its native environment. Thus, it can consider proteins both in water and a membrane setting, allowing for the design of protein-protein docking software for both of these distinct environments.

We have discussed how our novel scoring function attempts to maximise the surface complementarity of two STID map isosurfaces. The isosurface was defined through independent verification of the isovalue (0.43) in Section 2.3.4, but we also find that this choice of isovalue, on average, returns the highest score at the ground truth for both the water-soluble (Section 3.2.1.2) and transmembrane protein complexes (Section 3.4.2), despite the differing environments. This surface-based scoring function, shown to be smoother than traditional atomistic approaches (Figure 3.4.4), defines a potential energy surface which is then navigated by a particle swarm optimisation engine to predict and provide accurate arrangments. At the end of the optimisation, all candidate models, typically several thousand, are clustered and returned to the user (in this work, 300 solutions are returned). Applying our docking protocol to integral membrane proteins requires only the MD system setup to be altered, longer MD timescales, and a reduced search space. A pipeline for both docking procedures has been packaged and is available for free on GitHub through our protein-protein docking software, JabberDock (`https://github.com/degiacom/JabberDock`).

Applying JabberDock to a benchmark of 224 unbound, dimeric water-soluble protein complexes yielded an overall CAPRI-defined success rate $> 54\%$. Notably, the success of JabberDock was not adversely affected when it was challenged with the more difficult, flexible proteins, indicating the ability of the STID map to accommodate localised flexibility. With a smaller benchmark of 20 unbound dimeric, integral membrane proteins, JabberDock returned a successful model in 75% of cases, a remarkable improvement to the water-soluble benchmark. This places JabberDock as competitive in both protein-protein docking fields. Unsuccessful cases in both datasets were those which the optimiser struggled to navigate into due to obscured non-convex binding pockets, or, more often, those with a lack of surface features (*i.e.* flat) that the scoring function can use to designate good or bad poses. Addressing these two boundary conditions is the focus of the majority of the work in Section 4.4. We also hypothesise that using a STID map extracted from a protein's native bound complex, acting as a surrogate bound complex, could assist in unbound docking for both soluble and integral membrane protein docking. We explore this matter further in Section 4.3.

Following its success, we utilised JabberDock in two applications. The first aimed to predict starting dimeric structures for transthyretin fibril growth as a pre-processing step in a new auto-tiling algorithm. JabberDock was able to provide numerous, distinct complexes which corroborated known binding site positions. These initial periodic units were then utilised in the auto-tiling procedure, producing fibres that matched EM negative stain images. The second application looked into the feasibility that bo_3 oxidase could dimerise in a membrane, a result inferred from a recent employment of Mass Photometry to integral membrane protein structure characterisation. The structures generated indicated a mass of detergent bound to the dimer which matched the experimental Mass Photometry data, thereby confirming the dimerisation hypothesis. The success with the soluble and integral membrane protein benchmarks, and the results from these two applications, confirm the necessity of considering local dynamics during protein assembly, and consequently the applicability of JabberDock to a wide-range of biomolecular modelling scenarios.

While the STID isosurfaces have proven successful as a critical component of our docking method, there is a rich amount of structural, dynamic and electrostatic information held within a STID map. Future versions of JabberDock will explore the possibility of leveraging on this additional source of structural information to provide the user with more accurate models. Other areas of future investigation will include the adoption of different functions as a model for the pseudo-electron density discussed in Section 2.2 (*e.g.*, a Lorentzian), using JabberDock for rescoring models predicted by other protein docking methods, a reranking process building upon our preliminary results on the usage of a dipole complementarity score, the use of a different atomistic force field (including a polarisable one) to explore the impact on the STID maps, and an additional post-processing step based on MD or MC techniques to refine the best docking poses. In this context, we also foresee that the use of optimisation algorithms requiring no weighting could be beneficial.[40] Looking at the current clustering approach, our choice of the metrics used for K-means clustering may be suboptimal, as we are essentially giving an extra weighting towards the rotational coordinates. Indeed, as we see in Chapter 4, the translational coordinates are, in general, a larger determinant of success under the CAPRI criteria (see Section 3.2.3.2). A superior metric to perform K-means clustering with may be the RMSD variation within our initial solution pool. Therefore, future versions of JabberDock should change the metric used to cluster solutions, provided that it can return higher-quality docking solutions overall. Finally, while JabberDock performs well largely irrespective of case difficulty, there are limitations as we begin to consider larger conformational changes, allosteric response upon binding and so on. These are areas, critical to protein function, where the communities understanding and predictive abilities significantly fall short. In Chapter 4, we will discuss some of the early work being done to address some of JabberDock's limitations, with the goal of establishing JabberDock as one of the most competitive software available for structural biologists. Overall, enhancements in the scoring function and solutions reranking will help improve the performance of JabberDock against cases with low interface complexity, while refining the optimisation engine will reinforce its performance against cases with highly complex ones.

3.7 Bibliography

[1] N. S. Pagadala, K. Syed and J. Tuszynski, *Software for molecular docking: a review.*, *Biophys. Rev.*, 2017, **9**, 91–102.

[2] M. T. Degiacomi and M. Dal Peraro, *Macromolecular symmetric assembly prediction using swarm intelligence dynamic modeling.*, *Structure*, 2013, **21**, 1097–1106.

[3] M. F. Lensink, R. Méndez and S. J. Wodak, *Docking and scoring protein complexes: CAPRI 3rd Edition*, *Proteins Struct. Funct. Bioinforma.*, 2007, **69**, 704–718.

[4] M. T. Degiacomi, *Molecular Modeling of Bacterial Nanomachineries*, *Ph.D. thesis*, EPFL, 2012.

[5] M. C. Lawrence and P. M. Colman, *Shape Complementarity at Protein/Protein Interfaces*, *J. Mol. Biol.*, 1993, **234**, 946–950.

[6] T. Vreven, I. H. Moal, A. Vangone, B. G. Pierce, P. L. Kastritis, M. Torchala, R. Chaleil, B. Jiménez-García, P. A. Bates, J. Fernandez-Recio, A. M. J. J. Bonvin and Z. Weng, *Updates to the Integrated Protein-Protein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2*, *J. Mol. Biol.*, 2015, **427**, 3031–3041.

[7] H. J. Berendsen, D. van der Spoel and R. van Drunen, *GROMACS: A message-passing parallel molecular dynamics implementation*, *Comput. Phys. Commun.*, 1995, **91**, 43–56.

[8] I. H. Moal and P. A. Bates, *SwarmDock and the use of normal modes in protein-protein Docking*, *Int. J. Mol. Sci.*, 2010, **11**, 3623–3648.

[9] T. M.-K. Cheng, T. L. Blundell and J. Fernandez-Recio, *pyDock: Electrostatics and desolvation for effective scoring of rigid-body protein-protein docking*, *Proteins Struct. Funct. Bioinforma.*, 2007, **68**, 503–515.

[10] R. Chen, L. Li and Z. Weng, *ZDOCK: An initial-stage protein-docking algorithm*, *Proteins Struct. Funct. Genet.*, 2003, **52**, 80–87.

[11] C. Dominguez, R. Boelens and A. M. Bonvin, *HADDOCK: A protein-protein docking approach based on biochemical or biophysical information*, *J. Am. Chem. Soc.*, 2003, **125**, 1731–1737.

[12] J. T. Marinko, H. Huang, W. D. Penn, J. A. Capra, J. P. Schlebach and C. R. Sanders, *Folding and Misfolding of Human Membrane Proteins in Health and Disease: From Single Molecules to Cellular Proteostasis*, *Chem. Rev.*, 2019, **119**, 5537–5606.

[13] M. F. Lensink, S. Velankar and S. J. Wodak, *Modeling proteinprotein and proteinpeptide complexes: CAPRI 6th edition*, *Proteins Struct. Funct. Bioinforma.*, 2017, **85**, 359–377.

[14] N. Hurwitz, D. Schneidman-Duhovny and H. J. Wolfson, *Memdock: An α-helical membrane protein docking algorithm*, *Bioinformatics*, 2016, **32**, 2444–2450.

[15] P. I. Koukos, I. Faro, C. W. van Noort and A. M. Bonvin, *A Membrane Protein Complex Docking Benchmark*, *J. Mol. Biol.*, 2018, **430**, 5246–5256.

[16] S. Viswanath, L. Dominguez, L. S. Foster, J. E. Straub and R. Elber, *Extension of a protein docking algorithm to membranes and applications to amyloid precursor protein dimerization*, *Proteins Struct. Funct. Bioinforma.*, 2015, **83**, 2170–2185.

[17] M. A. Lomize, A. L. Lomize, I. D. Pogozheva and H. I. Mosberg, *OPM: Orientations of proteins in membranes database*, *Bioinformatics*, 2006, **22**, 623–625.

[18] D. H. De Jong, G. Singh, W. F. Bennett, C. Arnarez, T. A. Wassenaar, L. V. Schäfer, X. Periole, D. P. Tieleman and S. J. Marrink, *Improved parameters for the martini coarse-grained protein force field*, *J. Chem. Theory Comput.*, 2013, **9**, 687–697.

[19] I. Liko, M. T. Degiacomi, S. Mohammed, S. Yoshikawa, C. Schmidt and C. V. Robinson, *Dimer interface of Bovine cytochrome c oxidase is influenced by local posttranslational modifications and lipid binding*, *Proc. Natl. Acad. Sci. U. S. A.*, 2016, **113**, 8230–8235.

[20] A. Fiser and A. Sali, *ModLoop: Automated modeling of loops in protein structures*, *Bioinformatics*, 2003, **19**, 2500–2501.

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

[22] S. Schott-Verdugo and H. Gohlke, *PACKMOL-Memgen: A Simple-To-Use, Generalized Workflow for Membrane-Protein-Lipid-Bilayer System Building*, *J. Chem. Inf. Model.*, 2019, **59**, 2522–2528.

[23] J. P. M. Jämbeck and A. P. Lyubartsev, *Derivation and Systematic Validation of a Refined All-Atom Force Field for Phosphatidylcholine Lipids*, *J. Phys. Chem. B*, 2012, **116**, 3164–3179.

[24] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, *ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB*, *J. Chem. Theory Comput.*, 2015, **11**, 3696–3713.

[25] B. Hess, H. Bekker, H. J. Berendsen and J. G. Fraaije, *LINCS: A Linear Constraint Solver for molecular simulations*, *J. Comput. Chem.*, 1997, **18**, 1463–1472.

[26] M. Ramirez-Alvarado, J. W. Kely and C. M. Dobson, *Protein misfolding diseases: current and emerging principles and therapies*, Wiley, 1st edn., 2010, pp. 295–815.

[27] T. Coelho, *Familial amyloid polyneuropathy: New developments in genetics and treatment*, *Curr. Opin. Neurol.*, 1996, **9**, 355–359.

[28] D. R. Jacobson, R. D. Pastore, R. Yaghoubian, I. Kane, G. Gallo, F. S. Buck and J. N. Buxbaum, *Variant-Sequence Transthyretin (Isoleucine 122) in Late-Onset Cardiac Amyloidosis in Black Americans*, *N. Engl. J. Med.*, 1997, **336**, 466–473.

[29] C. Andrade, *A peculiar form of peripheral neuropathy: Familiar atypical generalized amyloidosis with special involvement of the peripheral nerves*, *Brain*, 1952, **75**, 408–427.

[30] P. Westermark, K. Sletten, B. Johansson and G. G. Cornwell, *Fibril in senile systemic amyloidosis is derived from normal transthyretin*, *Proc. Natl. Acad. Sci. U. S. A.*, 1990, **87**, 2843–2845.

[31] T. Eneqvist, K. Andersson, A. Olofsson, E. Lundgren and A. E. Sauer-Eriksson, *The β-slip: A novel concept in transthyretin amyloidosis*, *Mol. Cell*, 2000, **6**, 1207–1218.

[32] L. Cendron, A. Trovato, F. Seno, C. Folli, B. Alfieri, G. Zanotti and R. Berni, *Amyloidogenic potential of transthyretin variants. Insights from structural and computational analyses*, *J. Biol. Chem.*, 2009, **284**, 25832–25841.

[33] H. A. Lashuel, C. Wurth, L. Woo and J. W. Kelly, *The most pathogenic transthyretin variant, L55P, forms amyloid fibrils under acidic conditions and protofilaments under physiological conditions*, *Biochemistry*, 1999, **38**, 13560–13573.

[34] L. Fagerberg, K. Jonasson, G. Von Heijne, M. Uhlén and L. Berglund, *Prediction of the human membrane proteome*, *Proteomics*, 2010, **10**, 1141–1149.

[35] L. J. Catoire, X. L. Warnet and D. E. Warschawski, in *Membr. Proteins Prod. Struct. Anal.*, Springer New York, 2014, pp. 315–345.

[36] A. Sonn-Segev, K. Belacic, T. Bodrug, G. Young, R. T. VanderLinden, B. A. Schulman, J. Schimpf, T. Friedrich, P. V. Dip, T. U. Schwartz, B. Bauer, J. M. Peters, W. B. Struwe, J. L. Benesch, N. G. Brown, D. Haselbach and P. Kukura, *Quantifying the heterogeneity of macromolecular machines by mass photometry*, *Nat. Commun.*, 2020, **11**, 864553.

[37] I. Palombo and D. O. Daley, *Heme incorporation into the cytochrome bo3 occurs at a late stage of assembly*, *FEBS Lett.*, 2012, **586**, 4197–4202.

[38] A. Musatov, J. Ortega-Lopez, B. Demeler, J. P. Osborne, R. B. Gennis and N. C. Robinson, *Detergent-solubilized Escherichia coli cytochrome bo3 ubiquinol oxidase: A monomeric, not a dimeric complex*, *FEBS Lett.*, 1999, **457**, 153–156.

[39] A. Shrake and J. A. Rupley, *Environment and exposure to solvent of protein atoms. Lysozyme and insulin*, *J. Mol. Biol.*, 1973, **79**, 351–71.

[40] G. Tamò, A. Maesani, S. Träger, M. T. Degiacomi, D. Floreano and M. D. Peraro, *Disentangling constraints using viability evolution principles in integrative modeling of macromolecular assemblies*, *Sci. Rep.*, 2017, **7**, 235.

4 | Beyond JabberDock

4.1 Introduction

The version of JabberDock currently available for download on GitHub `https://github.com/degiacom/JabberDock` has shown itself to be competitive with respect to other protein-protein docking software. In particular, the integral membrane protein docking pipeline is a remarkable improvement over other methods. However, there is still clear room for improvement. Throughout Chapters 2 and 3 we have discussed the various challenges that hindered JabberDock and the concepts that could improve its success. These include:

1. Harnessing bound dynamics to assist in unbound docking.
2. Considering improved optimisation strategies to navigate ligands into narrow or occluded binding pockets.
3. Successfully docking isosurfaces with relatively flat binding interfaces (*i.e.* a lack of topographically distinct features), cases which the current scoring function struggles with.
4. Utilising a scoring function with minimal empirical weightings or terms to maximise transferability and reliability.
5. Reducing the overall computational costs associated with JabberDock such that the software is more accessible.

In this Chapter, we shall discuss our attempts to address these objectives.

We have already seen that the bound state impacts the topology of a STID map in Section 2.6. Therefore, we considered whether the use of a ligand's STID map extracted from a native bound complex could assist in unbound docking. We first discuss some preliminary results from the various benchmarks performed in Chapter 3, before testing the approach further with the BPTI ligand. Results demonstrate that leveraging bound dynamics as a surrogate for a target complex, does indeed improve unbound docking success rates.

Addressing objectives 2, 3 and 4 requires a little more creative thinking. Docking shapes, in our case, the protein isosurfaces, can be interpreted as a collision detection problem. Such problems are common in Computer Graphics research. Therefore, we will introduce techniques borrowed from this field to better represent a protein's STID map isosurface in a protein docking context, following the same methodology as Section 2.6.4.2. These protein representations allow us to leverage accessible gradients that directly lead to a binding site. We show that while an analytically derived scoring function works remarkably well for toy models, it proves inadequate for proteins due to the need to compensate for isosurface intersection. We, therefore, apply an adapted scoring function with an empirical term that accounts for this overlap. A significant amount of benchmarking was performed to guarantee both the highest quality results with respect to the CAPRI guidelines,[1] and to ensure we are sufficiently sampling the potential energy surface (PES). Particular focus is given to the clustering benchmarks employed to monitor solution convergence and diversity. While not complete, the results of this work inform us of the tasks yet to complete in the pursuit of JabberDock 2.0.

Finally, our endeavours to tackle the issues we encountered with our gradient-based method led to the design of a novel means to rapidly generate protein docked models for assessment. The approach borrows from ray-tracing algorithms, displacing a protein-ligand to a target receptor by the bounds of the distance between them. The result of this is an algorithm which is computationally cheaper than our previous efforts, addressing objective 5 above. This algorithm acts as a preliminary step in sufficiently sampling a non-trivial PES, thereby enabling some of the shortfalls of our gradient-based docking approach to be resolved.

4.2 Terminology

A lot of the techniques in this Chapter are borrowed from the Computer Graphics field, and thus come with their own terminology that may be unfamiliar to those in Biophysics. Here, we will briefly define this technical jargon in alphabetical order to give some context, but will avoid diving into any mathematical detail as it is beyond the scope of this work. The style of this is similar to the Common Terms & Abbreviations Section, but is given here for convenience.

Affine transformation A geometric transformation that preserves the original structural features of an object being transformed. In our case, these objects are the SDFs and the affine transformation refers to a roto-translation. It is equivalent to rotating and translating the ligand STID isosurface in Chapter 3. The affine grid is a means to apply these affine transformations to an SDF gridspace.

CUDA device A device which can support CUDA code, typically, although not exclusively, NVIDIA GPUs. CUDA is used as an interface to run code on the GPU in parallel, similar to MPI on a CPU.

Learning rate The amount the weights associated with a parameter are updated during optimisation iterations. For example, if the 3D Euclidean translation and the rotational quaternion are our parameters, the learning rate for each of these relates to how much each can be numerically changed by the optimiser during a step.

Loss Often referred in the context of a loss function, or cost function. In our work, it is effectively the energy associated with a specific arrangment at some arbitrary roto-translation. It is equivalent to the scoring function, but is usually taken as the inverse, *i.e.* instead of maximising the score, we minimise the loss.

Outer/inner-loop These are effectively nested *for* loops. Consider an algorithm which loops through the IMDb library of films to find the highest rated films of all time. The outer-loop may choose years to sample, while the inner-loop checks the films of that year and finds the highest rated ones. The outer-loop then stores the highest rated film of that year before moving on. In optimisation terms, the inner-loop performs some local optimisation, for example a gradient descent, while the outer-loop is more global. Outer-loop methods, such as Monte Carlo, are typically heuristic-based, and are able to rapidly sample large regions of the search space. Conversly, a gradient-descent method may be suboptimal globally, but are efficient at finding local minima.

Quaternion A 1×4 vector used here to represent a 3D spatial rotation, they are more compact and efficient than traditional rotational matrices. Throughout this work, we use exclusively unit quaternions, *i.e.* those normalised to one to avoid scaling.

Ray-tracing A rendering algorithmic technique which effectively traces paths of light from a source to a virtual object and simulates the scattering off that object. Ray-tracing is commonly employed in animated film-making and graphic design. Indeed, all protein images within this thesis have been ray-traced.

SDF Signed distance function. The signed distance function of a set provides the distance from some arbitrary point to the boundary of the set. For our work, the SDF of a point gives the distance to the boundary of the STID isosurface. The function has positive values inside the isosurface, and negative outside.

Unit sphere A sphere with radius 1. Most often used in conjunction with quaternions. In other words, the term “projecting the quaternion onto a unit sphere” means constraining the magnitude of the quaternion to 1. This prevents the optimiser from applying scaling factors to the ligand as an alternative means of increasing the score.

4.3 Harnessing dynamics from bound complexes for unbound docking

4.3.1 Preliminary work

Some proteins can form multiple complexes by interacting with different binding partners. Following our soluble and transmembrane protein docking benchmarks, we hypothesised whether knowledge of a protein's bound state with a specific partner might facilitate its docking with a different one, *i.e.* the bound state of the native complex from which the ligand or receptor is sourced can be used as a surrogate for the target complex.

From the water-soluble protein docking benchmark, an unbound case with structures that allowed us to test this theory is the UBA domain from Cbl-b ubiquitin ligase (PDB: 2OOA), a small 11 kDa protein. Although it is crystallised as a homodimer, it is the monomer that participates in the formation of a heteromultimer (PDB: 2OOB) with ubiquitin. Simulating a 2OOA monomer and using its associated STID map within JabberDock to predict the 2OOB complex yielded no successful results. On the other hand, generating a STID map using a monomer extracted from the simulation of its dimer gave intermediate quality results. This indicates that the dynamics of a Cbl-b ubiquitin ligase as part of a homodimer or a heteromultimer were similar. This similarity could be harnessed to improve the predictive power of our surface complementarity scoring. This approach was also tested with the significantly larger integrin I domain of complement receptor 3 complex (PDB: 4M76), but no good pose was found in this scenario. We subsequently applied the same assessment with the transmembrane wild-type cytochrome c oxidase from Rhodobactor sphaeroides (PDB: 1M56) test case, comprised of two binding partners (chains A and C) that have had their structures solved as part of an alternative complex. When docking STID maps generated from MD simulations of monomeric binding partners, none of the 300 candidate models were successful. In contrast, docking STID maps generated from surrogate bound-state conformations yielded 8 successful poses, the best one at rank 51. Although not featuring a top 10 successful result, this improvement indicates that membrane protein docking may also benefit from STID maps representing bound dynamics extracted from other known complexes these membrane proteins are involved with.

These cases certainly support our proposed hypothesis. To test this theory further, we require a protein which participates in a wide variety of complexes to measure the impact of extracted bound dynamics. As discussed in Section 2.6, the BPTI protein is prolific in the number of receptors it involved in (see Figure 4.3.1). Furthermore, we have already determined that the STID maps built from these different receptor environments are fundamentally different, reflected in the changes to the STID maps of individual amino acids. We will, therefore, apply these pre-generated STID maps from our previous work in Section 2.6 for the unbound BPTI ligand docking with the unbound matripase receptor (PDB: 1EAX), which corresponds to the benchmark case supplied by the CAPRI community.[2] If successful, this would predicate the usage of pre-existing bound complexes in generating accurate docked models of a target complex.

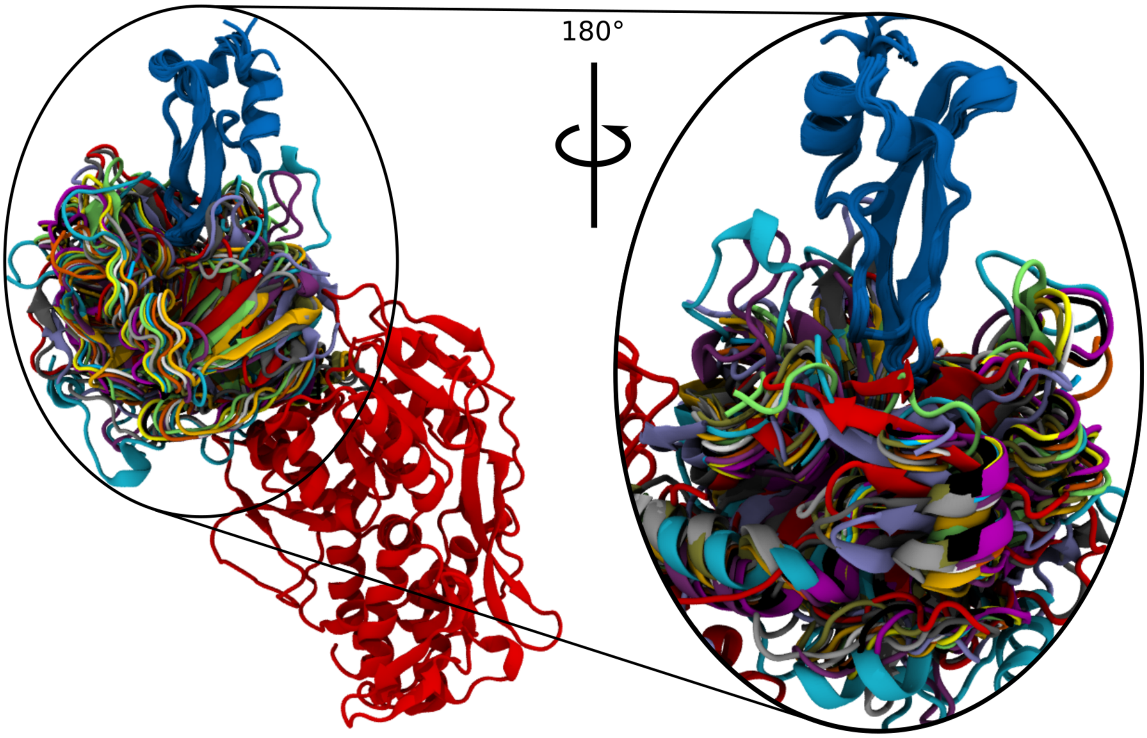

Figure 4.3.1: BPTI (blue) docked into all 15 receptors used in this work. The zoom inset is the same BPTI bound states rotated 180°, de-emphasising the presence of the genome polyprotein of Dengue virus 4 (PDB: 5YVU).

4.3.2 Results

Table 4.3.1 shows the results of docking a bound ligand extracted from its associated complex with the matripase unbound receptor. We use the CAPRI metrics to define success, and provide the associated quality of the closest prediction in the top 10 ranked results relative to the target structure of BPTI bound with matripase (PDB: 1EAW); acceptable, medium or high (see Section 3.2.3.2). Also provided for each, is the non-zero cross-correlation coefficient between the bound ligand STID maps and the unbound BPTI wild-type (PDB: 9PTI), as a measure of the difference in topology between the two. 1BPI is an unbound mutant variant of 9PTI with single point mutations: F22A, Y23A, N43G, and F45A.

Table 4.3.1: Quality of results from docking a bound ligand extracted from various complexes (given by the PDB code) into the unbound 1EAX receptor. The target structure, 1EAW, is highlighted in bold. Shown is the best RMSD in the top 10 ranked results, with associated $f_{\text{nat.}}$, RMSD_I as defined *via* the CAPRI criteria (see Section 3.2.3.2). The quality: incorrect (–), acceptable (*), medium (**) or high (***), of this predicted model is also provided. The results are ranked from highest quality prediction in terms of RMSD, to lowest. The wild-type 9PTI and mutant 1BTI are kept separate to denote the unbound structures.

PDB Code	CCC with 9PTI	RMSD / Å	$f_{\text{nat.}}$	RMSD_I / Å	Quality
1BTH	0.823	4.27	0.682	4.43	**
1EAW	**0.790**	**6.41**	**0.318**	**7.11**	*****
2R9P	0.854	6.70	0.364	7.44	*
2KAI	0.844	6.81	0.636	7.95	*
2RA3	0.850	6.93	0.409	8.41	*
1FY8	0.862	7.16	0.591	7.88	*
5YVU	0.833	7.55	0.773	7.31	*
1CBW	0.863	8.15	0.591	9.54	*
4WWY	0.814	8.40	0.864	8.82	*
1TPA	0.823	8.51	0.318	9.61	*
1YKT	0.850	8.67	0.727	8.39	*
2IJO	0.865	8.90	0.318	8.01	*
1BZX	0.857	8.98	0.273	8.60	*
3U1J	0.822	9.21	0.182	9.98	*
4DG4	0.825	10.00	0.409	9.05	–
9PTI	1.0	9.93	0.364	9.58	*
1BTI	0.836	16.72	0.000	16.21	–

Nearly all cases report a success, with the 1BTH ligand achieving an intermediate quality prediction. 4DG4, the only bound case without a success, lies on the boundary of an acceptable score with an RMSD of 10.00 Å. The unbound 9PTI BPTI ligand, while successful, is only narrowly an acceptable result with an RMSD of 9.93 Å, although this was the highest-ranked prediction. The unbound mutant 1BTI returned no successful models in the top 10 ranked results, indeed all top 10 models are very far removed from the known ground truth. The first acceptable prediction for 1BTI arrives at rank 34, with an RMSD of 7.94 Å, $f_{\text{nat.}}$ of 0.136 and RMSD_I of 10.73 Å. The average CCC for all BPTI with respect to 9PTI is 0.84(2), indicating that none of the STID maps are significantly topologically distinct from the rest of the set. However, we note that this scalar measure does not provide any information on local differences similar to Section 2.6, and the lack of any cases with CCC ≈ 1 does imply topological differences between maps.

4.3.3 Discussion

While the different receptors are distinct, all BPTI ligands are identical. The exception to this is the unbound 1BTI case, with four single point mutations. None of these amino acids participate actively in binding, although N44, which forms a hydrogen bond with R48 in matripase, is sandwiched between two of these mutations. Thus, despite the sequence of the ligand binding site and structure remaining identical, mutagenesis distal to the binding site has influenced JabberDock's likelihood to predict the known ground truth. The reasons for this are twofold:

1. The mutagenesis disrupts the internal hydrogen bonding network[3] of BPTI. Consequently, the dynamics of individual residues, including at the binding interface, are altered, as discussed in Section 2.6. Dynamics govern the STID map's topology. Thus, there are implications for subsequent docking.
2. The mutated residues will intrinsically alter the shape of the STID map isosurface, making some arrangements that were previously unfavourable more likely to occur. This is reflected through a low $f_{\text{nat.}}$, with different amino acids now found at the interface in the selection of best solutions.

These mutations highly destabilise BPTI,[3] thus rendering the formation of a complex with matripase unlikely. It is encouraging to see that JabberDock can predict the effect of mutants on protein-protein interactions.

Using the bound 1EAW ligand with the unbound 1EAW receptor (PDB: 1EAX), predictably returns one of the highest quality results. However, it is remarkable that the 1BTH case returns a better prediction overall. Following our work in Section 2.6, we know that the 1BTH receptor thrombin alters the dynamics of a few individual amino acids. It is interesting to see the repercussions of this, where these changes presumably enhance the quality of the predicted structure. Indeed, two of the top 10 cases were a medium success, in contrast to none in 1EAW; although, we note that 1EAW featured five overall successes in the top 10 versus 1BTH's four. While R42 in BPTI does not form a cation-π interaction with any residues in matripase, like it does with W44 in thrombin, 1EAW is the only complex other than 1BTH where BPTI's R42 is present at the binding interface. Indeed, the zoom inset of Figure 4.3.1 shows this, with 1BTH in cyan, and 1EAW in purple (see top left region of receptors interacting with BPTI). Thus, while the specific intermolecular bonds may be different, the transferability between 1BTH and 1EAW is greater than the other bound complexes, thereby enhancing the quality of the predicted pose.

In 13 of the 14 bound comparative cases, we return a higher quality prediction by harnessing the native-complex as a surrogate for the 1EAW bound receptor than simply using the unbound 9PTI ligand. Therefore, we have established that, despite the differences in the ligand STID maps imposed by the different receptors, there is some level of transferability between complexes. Given this transferability, it is unsurprising that the success rate is improved by this approach, since the bound cases attempted in the water-soluble benchmark in Section 3.3.1 overall did return a higher success rate. While using the bound ligand variants as opposed to the unbound did induce better quality results in the majority of cases, we emphasise that for the soluble 4M76 protein, no discernible improvement was observed. We initially considered whether this pre-processing step might only improve small protein docking, but our results with the transmembrane protein 1M56 indicate otherwise. The observed increase in docking accuracy from 1M56 is less significant than what we observed for the 2OOB and BPTI water-soluble globular cases, where the improvement yielded several successful complexes in the top 10 predictions. This difference is potentially because the impact on the dynamics at the binding interface is smaller when moving from a lipid environment to the protein surrogate *versus* water to the surrogate. Future work will need to quantify the statistical increase in success rate offered by performing this same benchmark on a much larger dataset of ligands with several distinct receptors, both for water- and lipid-soluble proteins. Regardless, results suggest that objective 1 in Section 4.1 is resolved, and that future versions of JabberDock, or enterprising users of the current version of JabberDock, should perform some form of homology checks prior to docking to determine whether there are any surrogate bound structures that could be used. By leveraging on these alternative complexes, JabberDock should yield more accurate predictions than anything found through pure unbound docking.

4.4 Gradient-based protein docking

4.4.1 The Signed Distance Function

The application of the STID isosurface in a protein-protein docking context has clearly proven competitive with respect to other methods. Its ability to accommodate flexibility and local electrostatics during the docking process provides a novel means to consider how two binding partners might view one another while interacting. However, there are some limitations with JabberDock, both from the optimisation and protein representation perspective:

1. Our current approach requires good empirical parameters: the distance cutoff (see Section 3.2.1.2), and various constants used in the scoring function (Equation 3.2.1) that are taken from the literature. These parameters may not be transferable to a much larger section of the proteome than our benchmark; itself extracted from the PDB, which as we discussed in Chapter 1, is weighted towards proteins that are easier to resolve with current techniques.
2. We can not be sure we are considering every possible arrangement, as there may be some locked behind high energy barriers which are missed by PSO's initial seeding process, or others that are trapped behind narrow pathways.
3. While we are exploring the PES with PSO, we never directly exploit accessible gradients towards a well-defined minimum at a binding site and instead rely on PSO inertia.
4. When considering an interaction between the receptor and ligand isosurfaces, we require several nested loops to measure just the score at a proposed interaction site, slowing the calculation down.

To address these concerns, we can reconceptualise the STID isosurfaces into something that is: (1) easier to manipulate, and (2) provides us with gradients for docking. For this, we borrow from techniques utilised in Computer Graphics for the development of animated films such as *The Incredibles* by Pixar.[4] Similar to Section 2.6.4.2, we convert our entire STID map isosurface into a Signed Distance Function (SDF), which we will denote as ϕ. Each point in space, λ, when acted upon by ϕ, returns the scalar distance to a zero-contour surface defined by the original STID isosurface. Points outside the object are negative, while those inside hold a positive value. This gives us direct access to information about the proximity to the surface while docking, and when considering the two SDFs, we achieve a surface complementarity maximum when these total distance fields are minimised. The SDF is differentiable almost everywhere, and its gradient satisfies the Eikonal equation:

$$||\nabla\phi|| = 1, \tag{4.4.1}$$

where $||\cdot||$ denotes the Euclidean norm. $\phi(\lambda) = 0$ is along the surface. Figure 4.4.1 shows an example SDF, both for a toy model and a protein's STID isosurface.

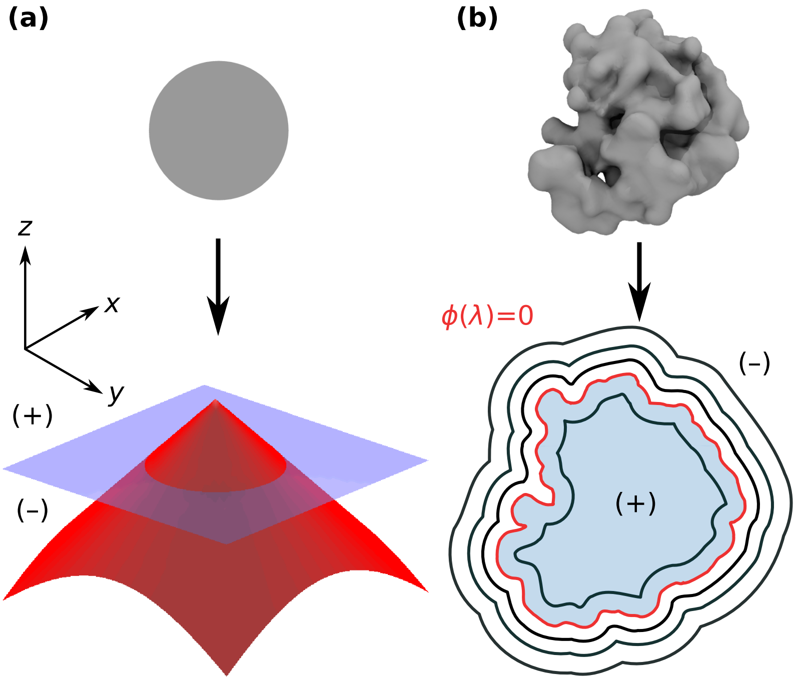

Figure 4.4.1: **(a)** 2D Disk (top) with associated exact signed distance function (bottom, in red).[5] The blue plane denotes the 2D coordinate system of the circle, while the z-axis relates to the value of the SDF at that coordinate. The intersection between the SDF and plane corresponds to the zero-contour surface. Therefore, positive SDF values are 'inside' the object, while negative SDF values are 'outside' the object. **(b)** Schematic of a STID isosurface of a protein (top) converted into an SDF (bottom).

4.4.2 The analytical solution – docking toy models

Let λ be a voxel point in space, where $\phi_1(\lambda)$ and $\phi_2(\lambda)$ return the SDFs for the receptor and ligand respectively. For an ideal case of docking with zero collision, we require a scoring function that is maximised when the receptor and ligand isosurfaces (where $\phi(\lambda) = 0$) have maximum surface complementarity. It should tend to 0 as the two are moved apart, and sharply decrease below zero to penalise any overlap. To obtain a function with these properties, we combine a Dirac function, $d(\lambda)$, that acts as a repulsive potential which increases sharply as the level of intersection increases, and a Heaviside function, $h(\lambda)$, which acts as an attractive potential, but asymptotes as we move further away from the surface. These are defined as follows:

$$d(\lambda) = \frac{1}{\pi}\frac{1}{(1+\lambda^2)} \; ; \; \lambda \in \mathbb{R}^{n\times 3}, \tag{4.4.2}$$

and

$$h(\lambda) = \frac{1}{2}\left(1 + \frac{2}{\pi}\right)\arctan(\lambda). \tag{4.4.3}$$

These two give the final energy for a single point λ:

$$e(\lambda) = \left|\, d(\phi_1(\lambda)) \times \big(h(\phi_1(\lambda)) + h(\phi_2(\lambda))\big) \,\right|, \tag{4.4.4}$$

where $\phi_2(\lambda)$ is the image of the ligand SDF following a projection into an affine grid which represents an associated roto-translation. $\phi_1(\lambda)$ is the SDF of the spatially-fixed receptor. The $|\cdot|$ denotes that we take the absolute value of the score. Since the overall score is the sum of all points, λ, the following integral is used:

$$E(\lambda) = \int_{\Omega} e(\lambda)\, \mathrm{d}\lambda. \tag{4.4.5}$$

Ω is taken over all points λ. A 1D example is shown in Figure 4.4.2.

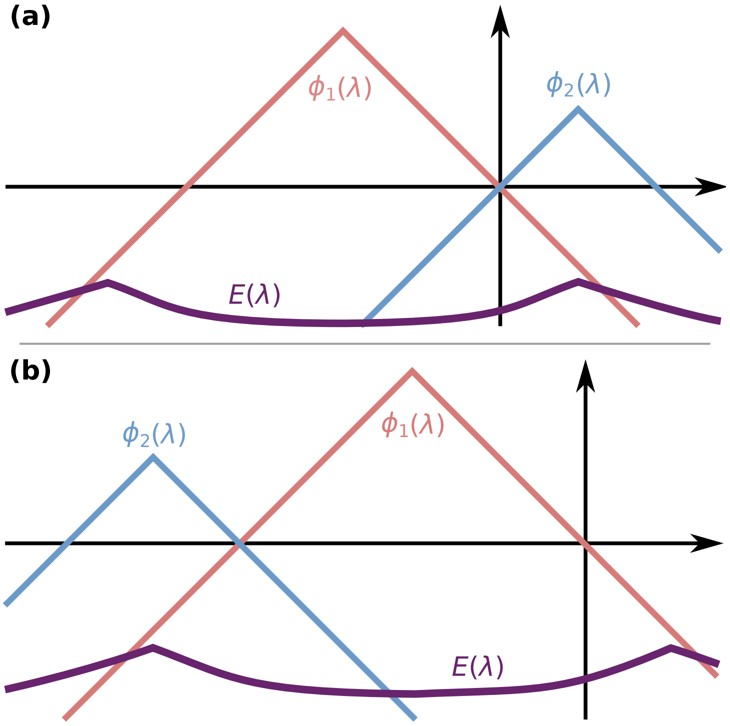

Figure 4.4.2: 1D SDF example of ligand (ϕ_2, blue) docking into the receptor (ϕ_1, red). Positive SDF values are inside the object (receptor or ligand), negative values are outside it, and zero at the edge. The two cases shown are those with object contact, but no intersection. Thus, the surface complementarity is maximised, corresponding to a maximum in the integral (palatinate) of Equation 4.4.5 at the two possible docked positions of ϕ_2. More specifically, the palatinate line relates to the value of the integral for each position of the ligand along the λ-axis (horizontal).

To exhaustively explore the search space associated with these scores, we initialise an ensemble of ligands by uniformly sampling a sphere fully surrounding the receptor.

Ligands are placed at each of these sampling points, and their position is then iteratively updated according to the gradient of our scoring function. Hereafter, these ligands, behaving as independent agents, will be named search particles. The specifics of initalisation and optimisation of search particles throughout Section 4.4 depends on the approach being taken, but all details are available in Section 4.4.6.

To test whether following the gradient of our scoring function allows a ligand to find its most suitable position on a receptor, we can expand our models to 3D, creating toy models for the receptor and ligand. We can conceive our receptor as a sphere with a spherical dimple, acting as a proxy for a binding site. The ligand is correspondingly a sphere with a radius matching the cavity. We discuss the creation of these toy models in Section 4.4.6.1. We initialise the position of 30 ligands through uniform sampling across the surface of some larger sphere around the receptor (Section 4.4.6.5). We then track the ligand's position as they follow the loss function gradient, where the loss function is the inverse of the scoring function, *i.e.* we're trying to minimise the negative of Equation 4.4.5, through the Adam optimisation algorithm (Section 4.4.6.3). We use the Adam algorithm available through the Geoopt Python package,[6] to keep our rotations normalised, thereby preventing any scaling. More specifically, we consider our spatial rotations as quaternions locked to the surface of a unit sphere, although in the context of the toy models the quaternion is irrelevent as the ligand is rotationally invariant. A visualisation of the optimisation process is provided in Figure 4.4.3.

To measure the success of our analytical scoring function, in conjunction with the SDF approach and Adam optimiser, we created two scenarios through two different receptor spheres. The first hosted a single binding site (top of Figure 4.4.3). The second had three potential sites of varying depth (bottom of Figure 4.4.3), with the target complex requiring the ligand to find its way into the largest site. This site was partly obscured by the entry point rim (*i.e.* it was larger than half the volume of the ligand), therefore requiring the ligand to physically pass through the receptor for a successful binding event. Figure 4.4.3 shows three stages for both of these cases, where the ligand, regardless of starting position, can successfully follow the gradient and dock into the correct binding site on the receptor. A short, 1-minute video is also available to watch these dockings in action, either by scanning the QR code or by following this link: `https://www.youtube.com/watch?v=lwXLIfgPf0O&t=1s`.

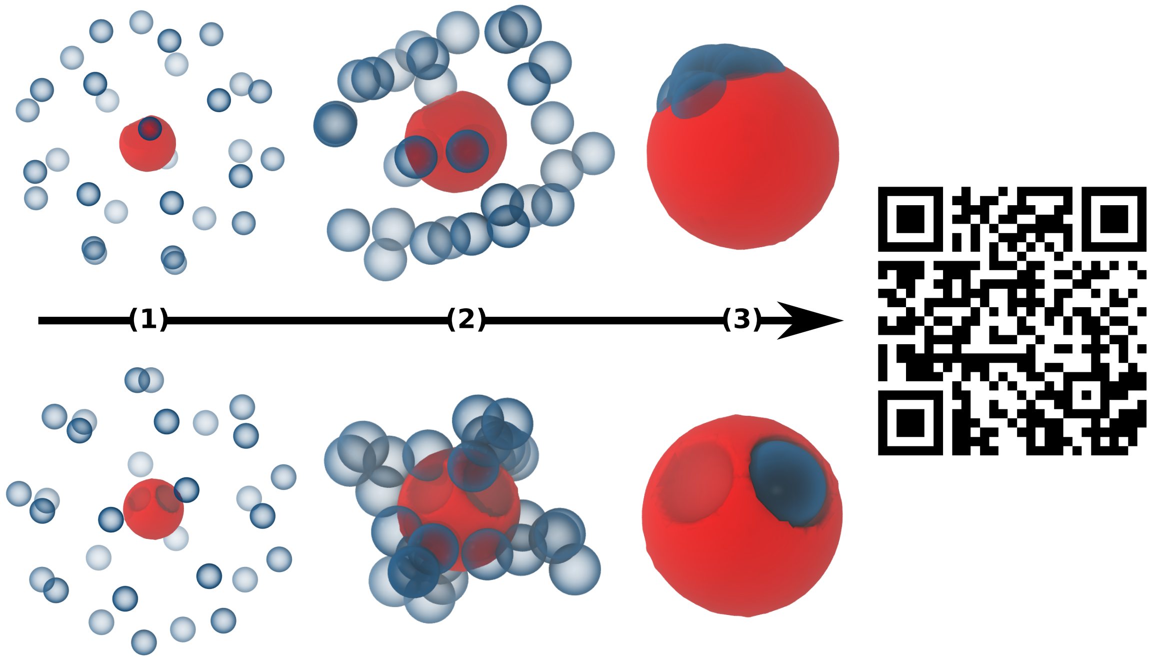

Figure 4.4.3: Three stages of the docking process for **(top)** single binding site receptor (red), and (bottom) multiple potential binding site receptor. **(1)** Ligands (blue) are initalised uniformly along the surface of some larger sphere around the receptor. **(2)** Ligands follow the gradient towards the binding site. **(3)** Regardless of starting position, the ligand is able to converge into the correct docking pose. Scan the QR code to watch a short video on these dockings, or follow this link: `https://www.youtube.com/watch?v=lwXLIfgPf0O&t=1s`.

By observing the video of the single-site case (Figure 4.4.3(top)), we note that the ligands oscillate around the binding site. This is likely due to the high learning rate over the short timeframe of the simulation, the learning rate being the rate at which the translation parameters are updated at each optimisation step. Of particular interest is the ligand's ability to pass through the receptor completely and emerge in the correct position. We can take a more detailed look into why the ligand takes a specific journey by exploring how the score relates to its spatial position, both along the receptor's surface and through it. Figure 4.4.4(a) demonstrates how the score changes as we roll the ligand along the receptor's circumference. Figure 4.4.4(b) shows how it changes as we pull the ligand through the centre of the receptor. Both the score and distance are in arbitrary units. There is a clearly defined minimum at the binding site, while the rim of the site and centre of the receptor have associated maximums with respect to the ligand's path. The rim maximum is much larger than the centre, explaining why the ligand prefers to pass through the receptor than move across its surface. The penalty for complete overlap is not too severe, which is unsurprising given the shape of the integral in Figure 4.4.2, and indeed is preferable than some surface locations on the receptor. This is ideal behaviour, as it would indicate that the boundary condition of JabberDock regarding the optimiser (objective 2 in Section 4.1) is resolved, although before confirming this we must first apply our analytical approach to the docking of real proteins.

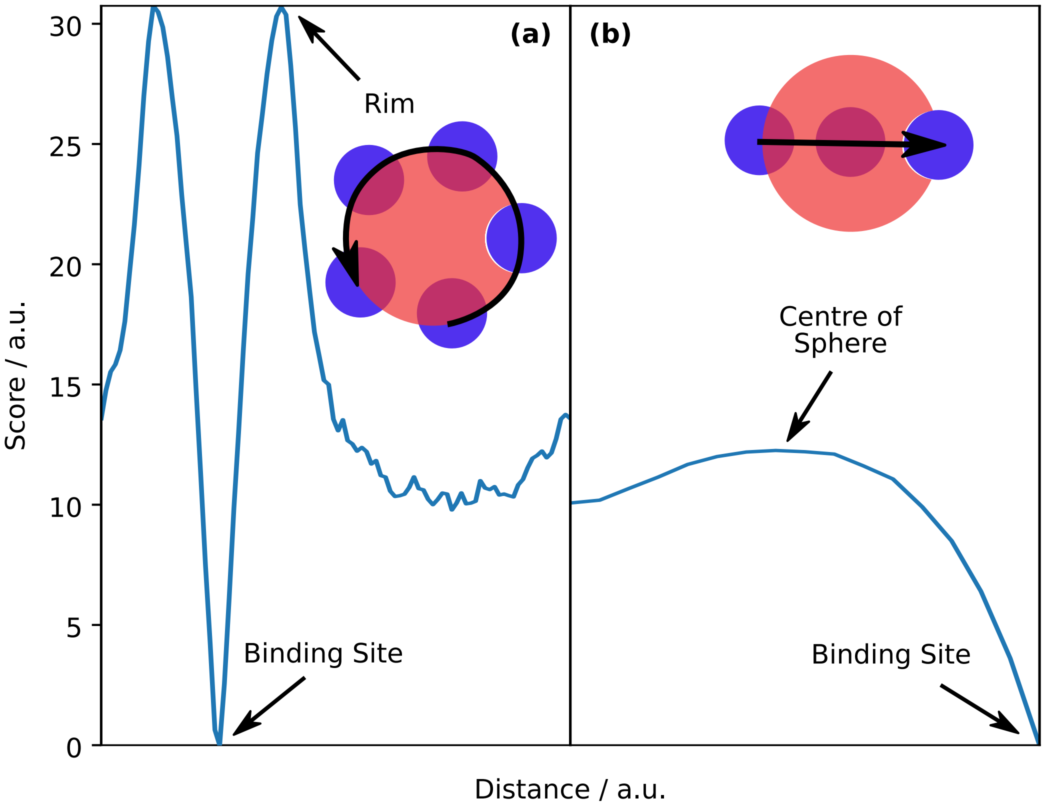

Figure 4.4.4: **(a)** Score returned by Equation 4.4.5 after rolling the ligand along the circumference of the receptor sphere toy model. **(b)** Score after pulling the ligand through the receptor. The distance is defined according to the respective schematic, with the global minimum in the score reflecting the correct docked pose. The score and distance are in arbitrary units.

4.4.3 The analytical solution – docking proteins

The toy models above are minimal examples to demonstrate the functionality of the scoring function in Equation 4.4.5. Their shape is homogenous except for the binding site positions, and although the multi-site sphere (bottom of Figure 4.4.3) has some greater level of heterogeneity, it is insignificant when compared with a protein's STID isosurface/SDF. The simplicity of the toy models, therefore, invites simplicity into the initialisation and optimisation of search particles. In real protein docking, there are three key factors that must also be considered. The first factor is our choice of isovalue. While our series of benchmarks in Section 2.3.4 determined that 0.43 was the best isovalue cutoff for the STID map isosurfaces, and indeed we showed in Section 3.2.1.2 that it performed the best for docking with JabberDock; due to the nature of the analytical scoring function penalising any overlap – where some overlap is inevitable due to the side-chain uncertainty, there may be an alternative choice of isovalue here that yields higher-quality results overall. The second factor is that we now require a 7-dimensional search space, to represent both rotations and translations; whereas previously the spherical toy model ligands were rotationally invariant. The third factor is that we now need to consider how to minimise the number of search particles, while also ensuring that the PES is thoroughly explored. In the toy models, the few docking sites induced a clearly defined global minimum for the search particles to converge into. The PES associated with real protein docking will have a significant quantity of distinct local minima, and while the ability of the ligand to pass through the receptor to reach the global minimum could offer some means to bypass energy barriers, the complexity of the PES means particles will still get trapped in local minima. Minimising the number of search particles required can be done through two ways: (1) optimising the starting positions of the ligand, and (2) determining the minimal density of search particles, or sampling granularity, required to sufficiently explore the PES. In optimising the starting positions, initialising the ligand positions on the surface of some sphere is suboptimal when docking into an elongated receptor, thus mapping these locations onto the surface of an ellipsoid is advantageous (see Figure 4.4.5 and Section 4.4.6.2). This is because it does not bias any search particles, and because it reduces the necessary number of them. In determining the sampling granularity, we consider both the number of sampling points on the ellipsoid, and the rotations applied to the ligands following their initial translation, which impacts the number of search particles per point on the ellipsoid.

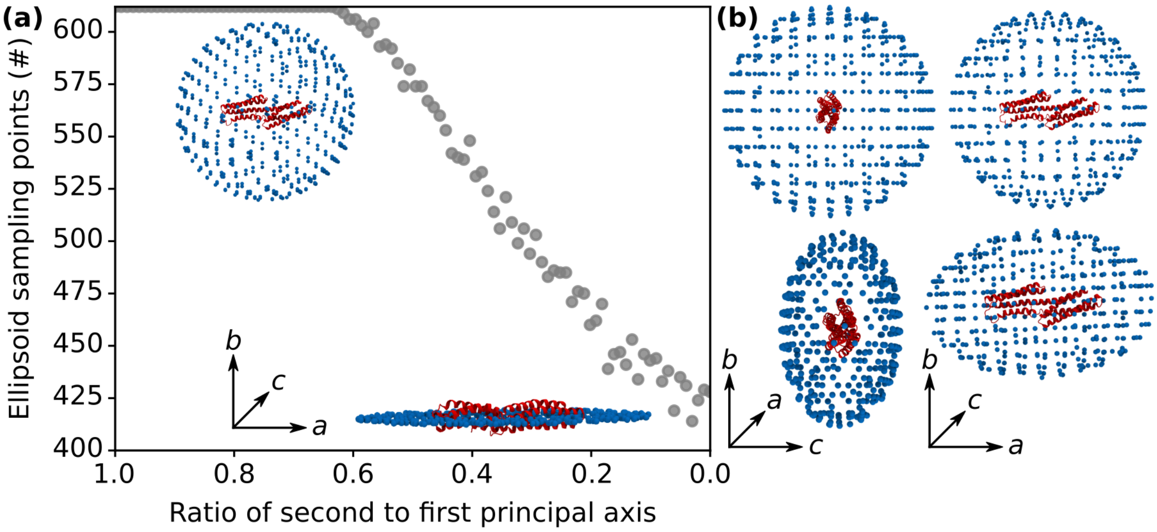

Figure 4.4.5: **(a)** Reduction in the number of sampling points from the ellipsoid mapping procedure, keeping the a- and c-axes fixed while reducing the b-axis. This moves the shape on which points are initialised from a sphere, through an ellipsoid, to a disc. The noise is due to the stochastic nature of the acceptance critera in Equation 4.4.13 (see Section 4.4.6.2). The initalised sampling positions are in blue, while the receptor (PDB: 1RKE) is in red. **(b)** Mapping of sphere sampling points (top) onto the surface of an ellipsoid (bottom), where $c < b < a$, following the procedure outlined in Section 4.4.6.2. This reduces the number of initial sampled positions from 612 to 429.

We addressed these three factors by performing a series of benchmarks to measure their impact. We first generated a series of SDFs between isovalues of 0.43 and 0.83 in steps of 0.04. We then initialised the starting positions of our search particles by uniformly sampling an ellipsoid determined by the receptor's shape (see Section 4.4.6.2) at three different sampling granularity levels in an approximately uniform fashion. Each starting position has two search particles associated with it, corresponding to the ligand spun 180° with respect to itself. Full details on the initialisation are available in Section 4.4.6.6. Following initialisation, each search particle is optimised until the loss converges (Section 4.4.6.3) to some local minimum, again by applying the Adam-based optimiser through Geoopt.[6] The converged translation and rotation are then applied to the atomic model, and the various criteria related to success measured (see Section 3.2.3.2). For this protein benchmark, we selected 19 complexes from the soluble protein benchmark[2] used in Chapter 3: 12 easy, 4 medium, and 3 hard (see Section 3.2.3.1 for information on difficulty classification). The diverse set of cases were chosen to test both our current method with previous successes, and to assess whether this SDF gradient-based approach could address the boundary conditions of JabberDock laid out through the objectives 2-4 in Section 4.1.

Preliminary tests showed that, while the toy ligands all converged to the global minimum, the proteins did not. Therefore, prior to deciding whether a full optimisation round was successful, we must cluster solutions to avoid redundancy. For this, we applied a DBSCAN-based[7] clustering algorithm (explained in Section 4.4.6.8). DBSCAN contains two key parameters: M and R, where M is the minimum number of points in a cluster, and R is the radius used to determine whether a point is close enough to a previously identified cluster to be included within it. M is traditionally used to discard noise, but since we wish to consider every possible converged solution, we set $M = 1$. We ascertain R through benchmarking. For this, we use two key measures:

1. The silhouette score.
2. The rand index between the DBSCAN identified clusters, and, given the number of clusters identified by the DBSCAN, a K-means equivalent.

The silhouette score is a measure of correctly assigned cluster diversity. It represents how similar a point in a cluster is to the other points in the cluster, compared with if it was a member of another cluster. It can range between -1 and 1, where positive values indicate the point is well-matched and cohesive. In contrast, a negative value implies that many points are poorly matched and that either the data is too random to be clustered, or there are too many or too few clusters. The rand index is a statistical measure to quantify the similarity between two data clusterings. A value close to 1 suggests the cluster sets are well-matched, while 0 is the opposite. We calculate a rand index between the clusters generated by DBSCAN, and then verify their validity with an independent clustering K-means method, using the same number of clusters determined by DBSCAN.

4.4.3.1 Effect of clustering on solution diversity

Figure 4.4.6 shows, for each protein and DBSCAN clustering radius R, across all isovalues and sampling granularity levels: (a) the average silhouette score, as a measure of the correct assignment to each cluster, and (b) the rand index between the DBSCAN clusters and K-means equivalent.

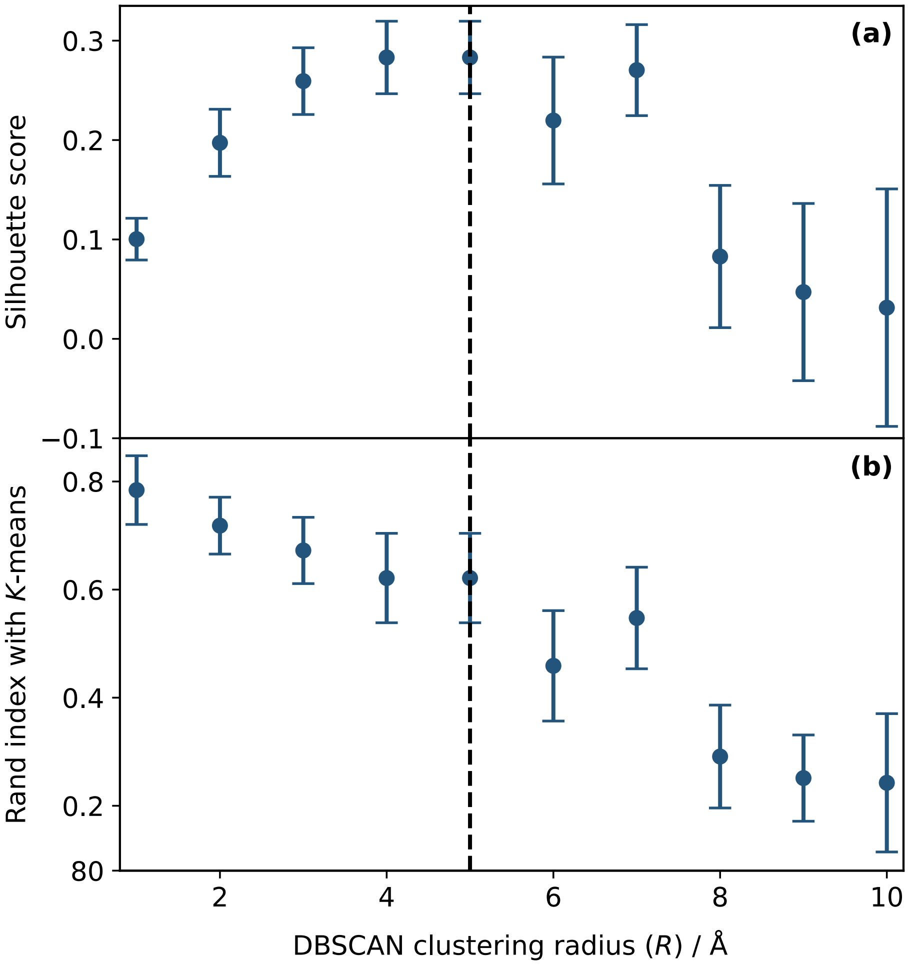

Figure 4.4.6: Identification of the ideal DBSCAN clustering radius, R, showing the variation of R with **(a)** silhouette score, **(b)** the rand index between clusters found with DBSCAN and a K-means equivalent with K set to the number of clusters given by DBSCAN. All values are averaged across all 19 benchmark proteins, isovalues and the three sampling granularity levels, with the standard deviation indicated by the error bars. The dotted line indicates the chosen value of R based on the results of this Figure.

Figure 4.4.6(a,b) shows that, to maximise the silhouette score and rand index, we require an $R \leq 5$. We then considered how the number of identified clusters, for each of the different sampling granularities, is dependent on R. Therefore, we calculated the average percentage increase in newly identified clusters from low to medium and medium to high sampling granularity levels for each R. The results of this are shown in Figure 4.4.7.

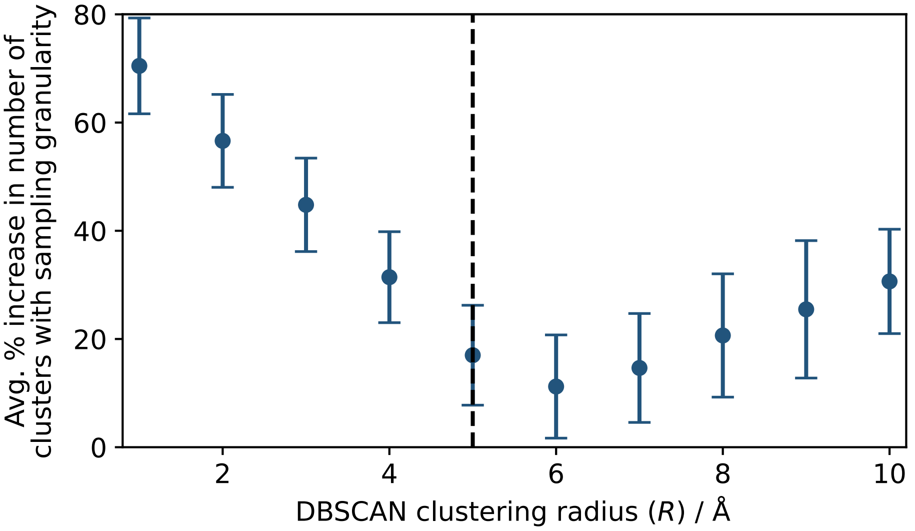

Figure 4.4.7: Identification of the ideal DBSCAN clustering radius, R, showing the variation of R with the average percentage increase in the number of identified clusters from low to medium and medium to high sampling granularity levels. All values are averaged across all 19 benchmark proteins and isovalues, with the standard deviation indicated by the error bars. The dotted line indicates the best choice of R based on this information and Figure 4.4.6.

The % increase in clusters in Figure 4.4.7 represents the average number of new clusters identified as we increase the number of search particles with sampling granularity. A larger percentage would indicate that newly added particles converge to novel solutions. It is expected for the small R to return a large increase, as they are essentially near-single-point clusters. Ideally, this % increase should be as small as possible for a range of R to confirm some convergence level in the sampling of the PES. Following the results from Figure 4.4.6, where we identified $R \leq 5$, we see here that a minimum increase in the number of new clusters of 17% is given when $R = 5$. This percentage increase indicates that the PES associated with Equation 4.4.5 has a high Lipschitz constant. In other words, the frequency of the PES is high, with many local minima in close proximity to one another. We should, therefore, extend to higher sampling granularity levels with a larger number of search particles. However, the computational cost of these calculations is significant, with a mean total optimisation time of 10 hours on an NVIDIA GeForce RTX 2080 Ti. GPU at the highest sampling granularity level of 12 (466 average search particles). We, therefore, opted to analyse the success of our current results and consider possible faster methods for later employment (see Section 4.5). We display the three levels in Figure 4.4.8 for completeness, and to monitor whether there is a shift in quality because of additional solutions. Within each cluster returned by $M = 1$ and $R = 5$, we checked whether further clustering was required to accommodate possible rotations. In every cluster across all isovalues and proteins, the quaternions converged to some fixed rotation, demonstrating the presence of clear gradients to discrete solutions. Thus, no further clustering was required.

4.4.3.2 Effect of isovalue on docking success rate

Given our choice of M and R, we can now measure the success of the clustered solutions, averaged across the 19 proteins, for each STID isovalue. Figure 4.4.8 compares these results with the equivalent success found for these 19 proteins with JabberDock.

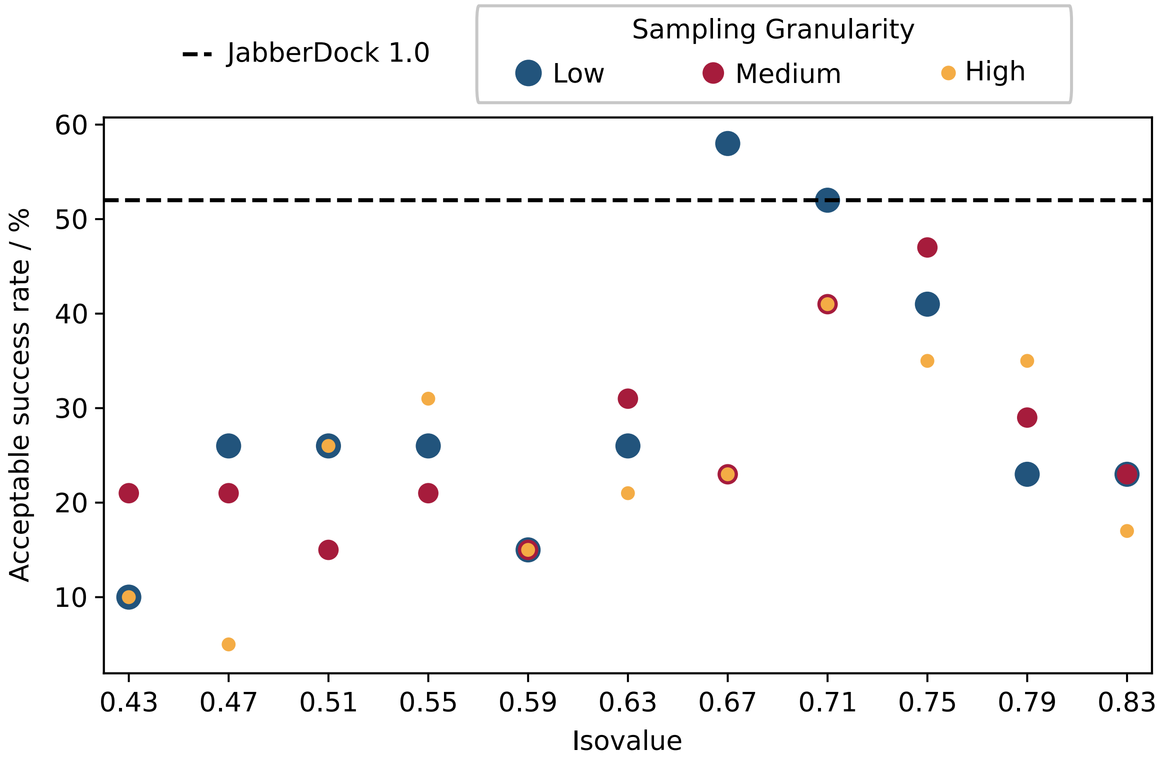

Figure 4.4.8: Acceptable success rate of the 19 protein complexes used in this analytical benchmark as a function of the isovalue used to generate the SDF. The dotted line provides the equivalent success for these 19 proteins in JabberDock (*i.e.* not the entire benchmark). Also provided are the different results returned by the different sampling granularity levels.

Figure 4.4.8 shows a significant difference in success between the sampling granularity levels. For example, an isovalue of 0.67 produces a successful model in 58.8% of cases at low sampling granularity, but increasing the number of search particles reduces this success significantly to 23.5%. In contrast, at an isovalue of 0.79 we obtain a success of 35.3% for high sampling granularity *versus* for 23.5% for low. Furthermore, there is a lack of consistency in the increase or decrease of quality with sampling granularity. For example, an isovalue of 0.63 at medium sampling granularity outperforms the other two. Clearly, there are regions of the PES with deeper wells that are not mapped with a smaller number of search particles despite clear gradients, confirming the complexity of the search space relative to the toy models.

In terms of medium quality success, a 5.2% success rate was found at an isovalue of 0.43 and low sampling granularity, and 5.2% at 0.75 and medium sampling granularity. 11.8 % of these cases returned a medium quality prediction with JabberDock. All successful results, where they occur, mirror those correctly predicted by JabberDock, which is encouraging as it demonstrates that this novel approach carries one of JabberDock's key strengths; its ability to accomodate local protein flexibility in docking. Where this gradient-based approach consistently outperforms JabberDock across all the isovalues trialled is the complex formed between a Fab fragment of a monoclonal IGG antibody and the major allergen from birch pollen Bet v 1 (PDB: 1FSK). This cases features a small, relatively flat binding site, suggesting that this gradient-based approach could overcome the scoring function boundary condition of JabberDock (objective 3 in Section 4.1).

The GAP domain of the Pseudomonas aeruginosa ExoS toxin and human Rac (PDB: 1HE1), is a case that was successful across all the different sampling granularity levels and isovalues. Here, we observe that many of the binding site interactions take place *via* intermolecular hydrogen bonding through water, as opposed to direct side-chain interactions. As we discussed in Section 2.5, our STID map, at an isosurface cutoff of 0.43, effectively represents an excluded volume. Docking 1HE1 would, therefore, require no volumetric overlap between isosurfaces, and still return a maximised surface complementarity.

Cases that feature a greater number of direct side-chain interactions, such as that of the cysteine desulfurase CsdA and sulfur-acceptor CsdE complex (PDB: 4LW4), tended to perform poorly with our analytical method. These complexes featured some level of STID isosurface overlap at the ground truth. While in JabberDock overlap was penalised, if it facilitated overall better surface complementarity, it was permitted. Here, the conditions are much stricter, preventing any form of SDF intersection in a converged solution, although as we saw in Section 4.4.2, the ligand may still pass through the receptor to dock into the binding site. The inability to converge into a solution featuring some intersection may prevent the optimiser from finding good poses, and explains why smaller isovalues, where side-chain uncertainty is represented explicitly in the topography of the STID isosurface, perform poorly. Further evidence for this is that, when placed in the known ground truth, in most cases, the ligand navigated away from the site outside the range of a successful model. Given the success of the higher isovalues, as well as the encouraging results from the toy models and this protein benchmark regarding a resolution to the boundary conditions of JabberDock, it may now be necessary to consider a means to permit some overlap in the scoring function to further increase the success rate.

4.4.4 The semi-empirical solution – docking proteins

While it is ideal to keep the scoring function purely analytical, to accommodate the necessary side-chain uncertainty, we must include an empirical parameter that accounts for the intersection. For this, we turn to a semi-empirical solution. To compensate for this intersection, and address the issue identified by the clustering, the high frequency

PES, we define a new scoring function for a point λ, with characteristics similar to that of a Lennard-Jones potential:

$$f(\lambda', k) = \frac{(\phi_1(\lambda') + k)\exp\left(-\frac{\phi_1(\lambda')+k}{k}\right)}{k^2} \ ; \ \lambda' \in \mathbb{R}^{n\times 3}. \tag{4.4.6}$$

It uses a sliding hyperparameter, k, that alters the maximum position of the integral such that overlap between the SDFs is feasible:

$$F(\lambda', k) = \int_{\Omega} f(\lambda', k)\ \mathrm{d}\lambda' \ . \tag{4.4.7}$$

λ' is a ligand boundary point (*i.e.* the vertices that define the surface of the ligand SDF) sampled in the receptor SDF. Any λ' points sampled outside the bounds of the receptor grid are set to ∞. Ω, therefore, represents every point within the bounds of the receptor grid. This is in contrast to Equation 4.4.5, where the integral sampled the receptor and ligand image SDFs individually. $k > 1.0$ permits overlap, and $0.0 < k < 1.0$ pushes the ligand further away from the binding site. When $k = 1.0$, the function is analytically exact for models where maximised surface complementarity is desired. This k parameter effectively accommodates for the uncertainty in the side-chain motion; the larger it is, the more intersection is permitted. Figure 4.4.9 shows a visual example for this for the STID maps of HLA-A2 complexed with the T cell coreceptor CD8 (PDB: 1AKJ), with increasing values of k leading to greater levels of molecular overlap. k has a significant impact on the overall score and subsequent exploration of the PES through our chosen optimiser and must be accurately determined through benchmarking. A 1D example of this function with two values of k is shown in Figure 4.4.10. It is possible to modify Equation 4.4.5 to include hyperparameters that permit overlap. However, both the Heaviside and Dirac subfunctions would require their own individual hyperparameter, whereas Equation 4.4.7 simplifies matters by only needing one.

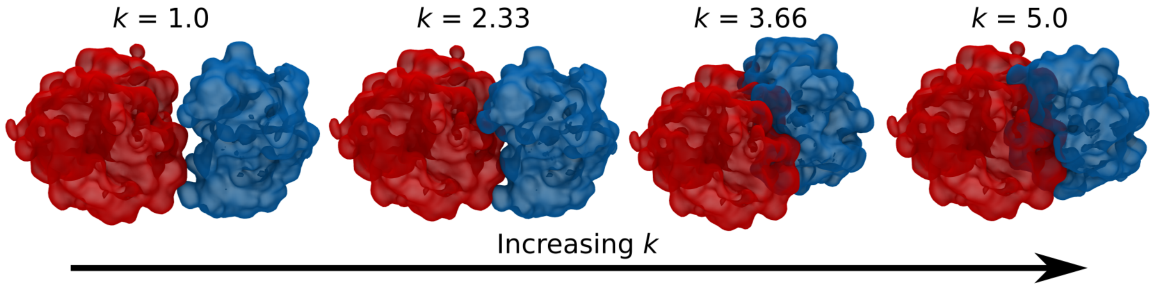

Figure 4.4.9: Converged docked result from the same initialised position for the HLA-A2 bound to the T cell coreceptor CD8 complex (PDB: 1AKJ). The ligand is shown in blue, the receptor in red. With the increasing value of k, we generate structures with greater levels of molecular overlap.

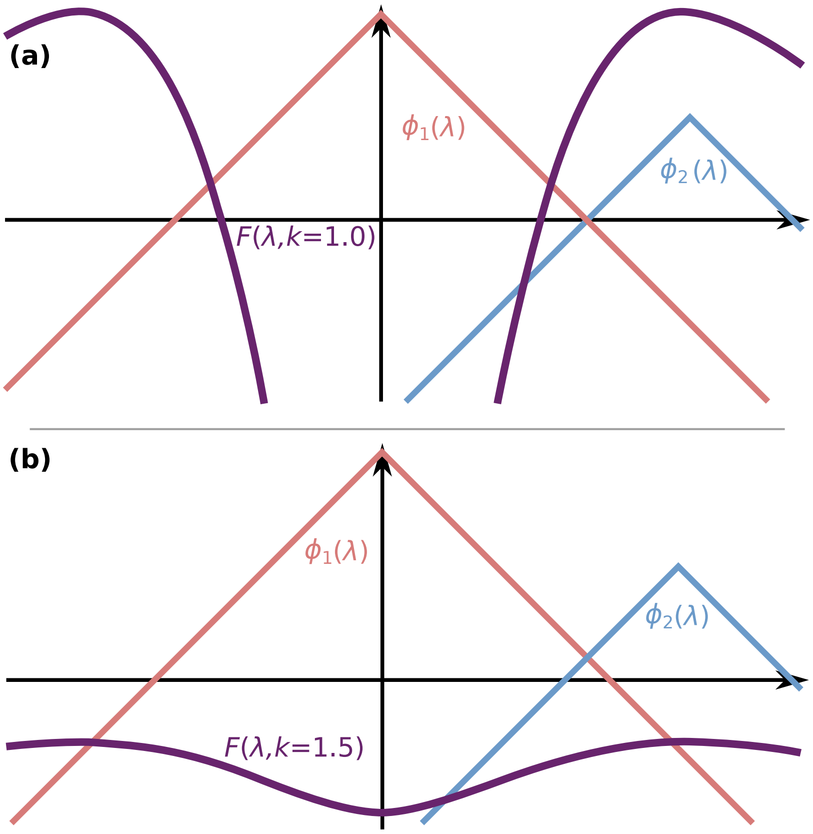

Figure 4.4.10: 1D SDF example of ligand (ϕ_2, blue) docking into the receptor (ϕ_1, red). Positive SDF values are inside the object (receptor or ligand), negative values are outside it, and zero at the edge. **(a)** $k = 1.0$, the analytically exact solution with no SDF overlap permitted, with the value of the integral (Equation 4.4.7), for each possible position of the ligand shown in palatinate. **(b)** $k = 1.5$, the maximum in the integral is shifted to inside the the receptor SDF.

4.4.4.1 Determining the overlap hyperparameter k

To determine the value of k, we begin with a parameter sweep of both k and the isovalue. For this benchmark, we selected a subset of 9 proteins used for the analytical benchmark; 6 easy, 1 medium and 2 hard. The reason for this reduction in the dataset was the execution times of our analytical dataset, with computational costs becoming more expensive due to the parameter sweep. Similar to the analytical benchmark, converged roto-translations are applied to the atomic models to assess the quality according to the CAPRI criteria.[1]

Our sweep ran through isovalues ranging between 0.4 and 0.72 in steps of 0.04. Recall, though, that we wish to compensate for side-chain uncertainty with k, not a higher isovalue. We swept through k between 0.5 and 5.0 in steps of 0.5. While a gradient descent based method is preferable with our SDF-based method with Equation 4.4.7, given this benchmark requires 72 individual optimisation rounds for 9 proteins, using Adam in series is not possible due to the time constraints. Therefore, we adopted a different sampling strategy, whereby we first performed a parameter sweep with particle swarm optimisation (PSO, see Section 4.4.6.4). The goal of this was to identify a region of interest in the parameters, to be further refined *via* Adam.

For the PSO, we utilised N search particles consisting of random translations mapped onto the surface of an appropriate ellipsoid (see Section 4.4.6.2) and a random rotation. We initially set N to 1000, but due to the ellipsoid mapping, N is reduced by a factor dependent on the shape of the receptor. 500000 iterations were performed in total per PSO round, with 75 rounds in total, each with a novel seeding of search particles. Thus, for each protein we perform $50000 \times 75 \times N$ individual evaluations. The solutions from the 75 individual rounds were then collated, and the top 10 scoring predictions with an RMSD difference of at least 5 Å between one another recorded. Full details for the PSO are given in Section 4.4.6.4. The ground truth was also scored at each value of k for comparison. Ideally, even if none of the top 10 predicted complexes for each protein are successful models, the ground truth score should at least fall within the margin of the top 10 scores. If this were the case, it would indicate that a superior optimisation strategy could natively find the ground truth. Figure 4.4.11 therefore compares, for each value of k, the ground truth score and the score of the 10th ranking model.

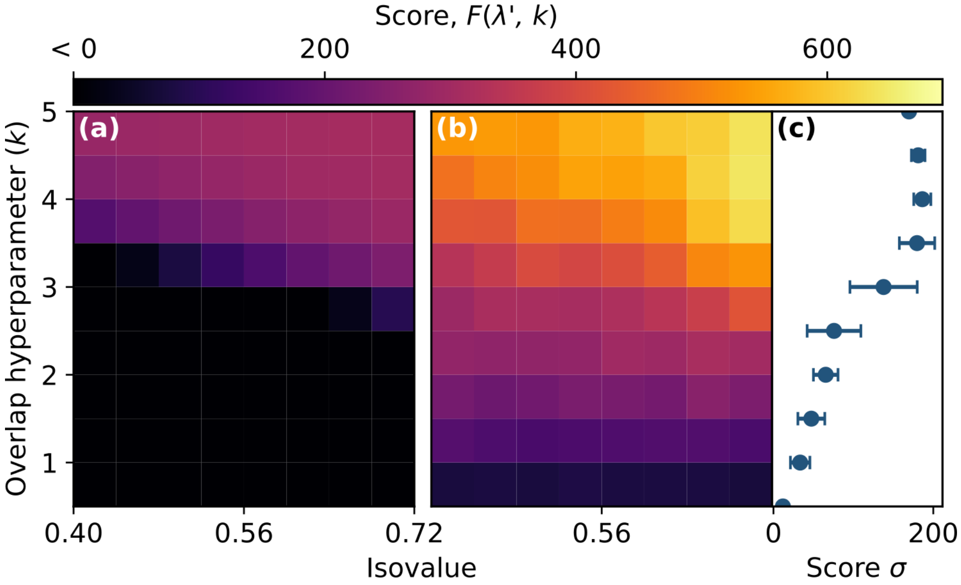

Figure 4.4.11: For each value of the hyperparameter determining the level of permitted overlap, k, and isovalue, the corresponding score, averaged across the 9 proteins, of the: **(a)** ground truth, **(b)** the 10th solution. **(c)** Standard deviation of the 10th model's score across the 9 proteins. The errorbars indicate the standard deviation across the different isovalue.

Figure 4.4.11(c) shows that there is little variation in the value of the score between proteins, and that the averages reported in 4.4.11(b) are not biased by some exceptionally high or low score from a single protein. The ground truth score is consistently smaller than the 10th ranked solution. Indeed, for large regions of both k and choice of isovalue, the ground truth returns a negative score overall, implying that we would never naturally discover it. However, since we are permitted to move some distance from the known ground truth under the CAPRI criteria,[1] we can also explore whether any of these top 10 models correspond to a success. Figure 4.4.12 shows the average success rate of the 9 proteins for each k and isovalue. For comparison, JabberDock returned a 44.4% success rate for this dataset.

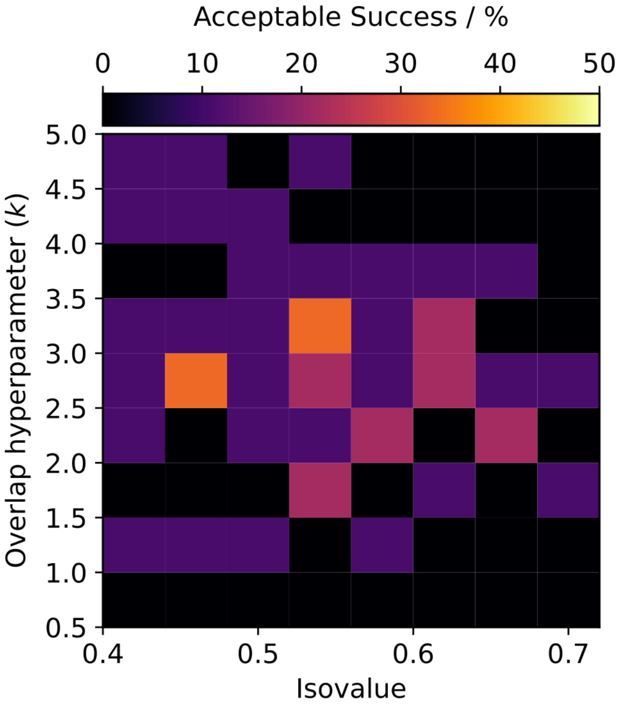

Figure 4.4.12: Average variation in acceptable success rate across the 9 proteins for each overlap hyperparameter, k, and isovalue choice.

We can see from Figure 4.4.12, that the highest quality models were returned between a k of 2.5 and 3.5, with a corresponding isovalue between 0.44 and 0.52. Therefore, we can use this to inform us of a starting point for k in a round of meta-optimisation. Considering that we wish to retain a lower isovalue to ensure consistency with our earlier work, we take the success peak at an isovalue of 0.44 to derive the SDF from the STID map in the next stage of our benchmark.

For this meta-optimisation stage, we return to Adam due to the reduced size of the parameter space. We treat k as a hyperparameter, optimised *via* an outer-loop Adam optimisation approach (Section 4.4.6.3) with the RMSD relative to the ground truth acting as a loss function. For each outer-loop iteration, we then ask an inner-loop Adam optimisation round to find the converged translations and rotations that yield the highest scores at the current value of k. Each of these scoring evaluations consists of a full docking run, where we scatter 300 ligand positions uniformly on the surface of some sphere around the receptor, map them to an ellipsoid, provide each search particle with a random rotation (see Section 4.4.6.7), and then optimise until convergence (see Section 4.4.6.3). The loss function of this inner-loop is set to Equation 4.4.7. The full meta-optimisation process was applied to each of the 9 proteins in the current benchmark. The converged k that returned the best score at the ground truth for each are given in Table 4.4.1.

Table 4.4.1: Converged value of the overlap hyperparameter k for each of the 9 proteins *via* meta-optimisation with respect to the ground truth. The number of iterations required for k convergence for each protein is also given.

PDB Code	k	Iterations #
1AKJ	3.58	36
1BVK	4.72	30
1EAW	3.29	12
1F6M	3.89	19
1FSK	4.55	27
1HE1	3.85	33
1IJK	5.38	55
1NSN	4.57	29
1RKE	3.61	9
Average	4.16	27.8

4.4.4.2 Effect of clustering on solution diversity

Applying an average k of 4.16, we then performed five repeat rounds of Adam optimisation with 200 particles each, once again initialised by scattering uniformly onto the surface of some sphere, then mapped onto an ellipsoid encompassing the receptor, before a random rotation was applied (see Section 4.4.6.7). These five additional optimisation rounds were used to check for solution convergence, similar to the sampling granularity check in Section 4.4.3. We, therefore, followed a similar procedure to the analytical benchmark, where we calculate the percentage increase in the number of additional clusters for each DBSCAN clustering radius R (see Section 4.4.6.8) between 1 Å and 10 Å, the results for which are given in Figure 4.4.13.

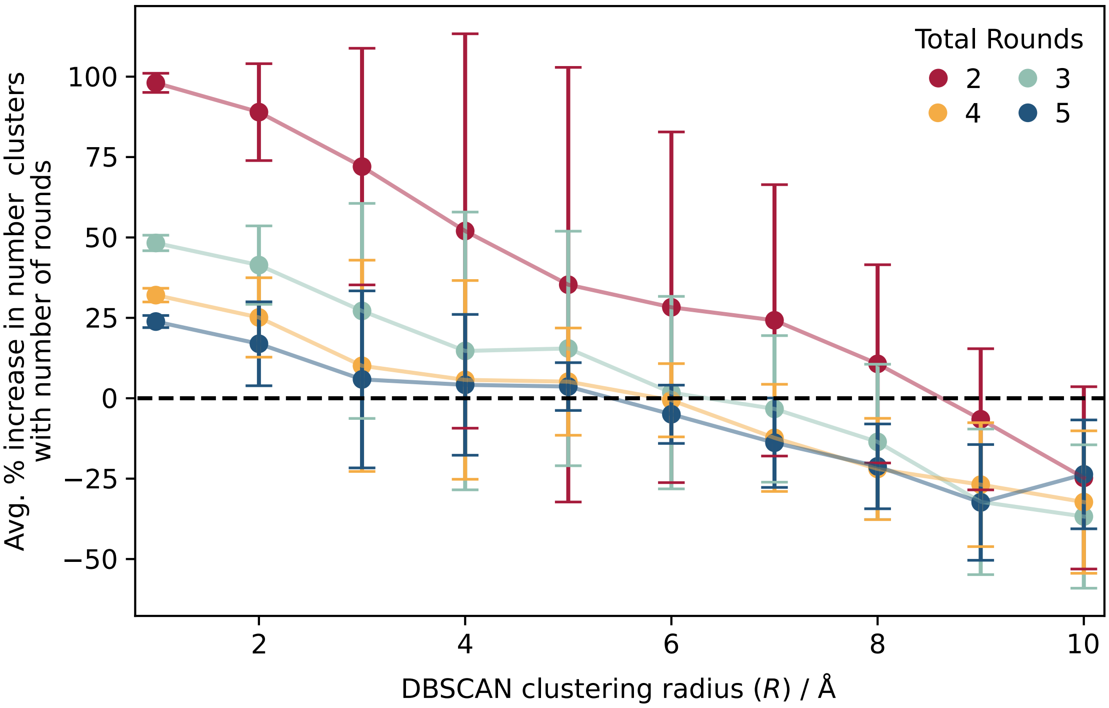

Figure 4.4.13: Identification of the ideal DBSCAN clustering radius, R. We show the dependency on R of the average percentage increase in the number of clusters identified *via* DBSCAN with each optimisation round for the empirical SDF gradient-based scoring function. Optimisation round refers to the total number of rounds (reseeded initialisation and optimisation) performed prior to a final clustering. The % increase is averaged over the 9 proteins used in this benchmark, with the error bars providing the standard deviation. The horizontal dotted line denotes that no additional clusters were identified. Ideally points should remain on this line; above it, and we are identifying novel clusters with each round – demonstrating that we are not sufficiently exploring the PES, below it suggests the presence of cluster merging events, removing diverse solutions.

Similar to Figure 4.4.7, we find in Figure 4.4.13 that at very low R, we continue to identify novel clusters with each round of optimisation; thus implying a lack of convergence. At very large R in Figure 4.4.13 we begin to decrease the number of identified clusters with increased number of search particles, in contrast to Figure 4.4.7. This implies that the converged data is not well clustered, as there are many points bridging spatial gaps between previously separated clusters, merging them into one. Indeed, after all five optimisation rounds, we find that at $R = 10$, we have only one cluster. We accordingly look for points in Figure 4.4.13 that are close to the zero-increase line without an error bar crossing it; to ensure that cluster merging events do not make obsolete high-quality solutions, or bundle high-quality solutions with those where the ligand is far from the binding site. Clearly, 200 individual randomly scattered particles are not enough to sample the space, and it is only when we get to the fifth round (totalling 1000 search particles), that we begin to achieve some convergence at low R. Similar to the analytical benchmark, a larger number of search particles are required here to secure convergence. Once again, however, we find that the calculations take a significant amount of time on the same hardware. 1000 total search particles corresponded to an average time of completion of 27 hours. Thus, given the *a posteriori* quality of the solutions, coupled with the length of time for a calculation, we instead opted to consider a superior sampling method (see Section 4.5).

To quantify the success of our current results, we took all five repeat rounds and clustered them using the same method as Section 4.4.3.1. In other words, we calculated our silhouette score and rand index for all sets of clusters at each R (see Figure 4.4.14).

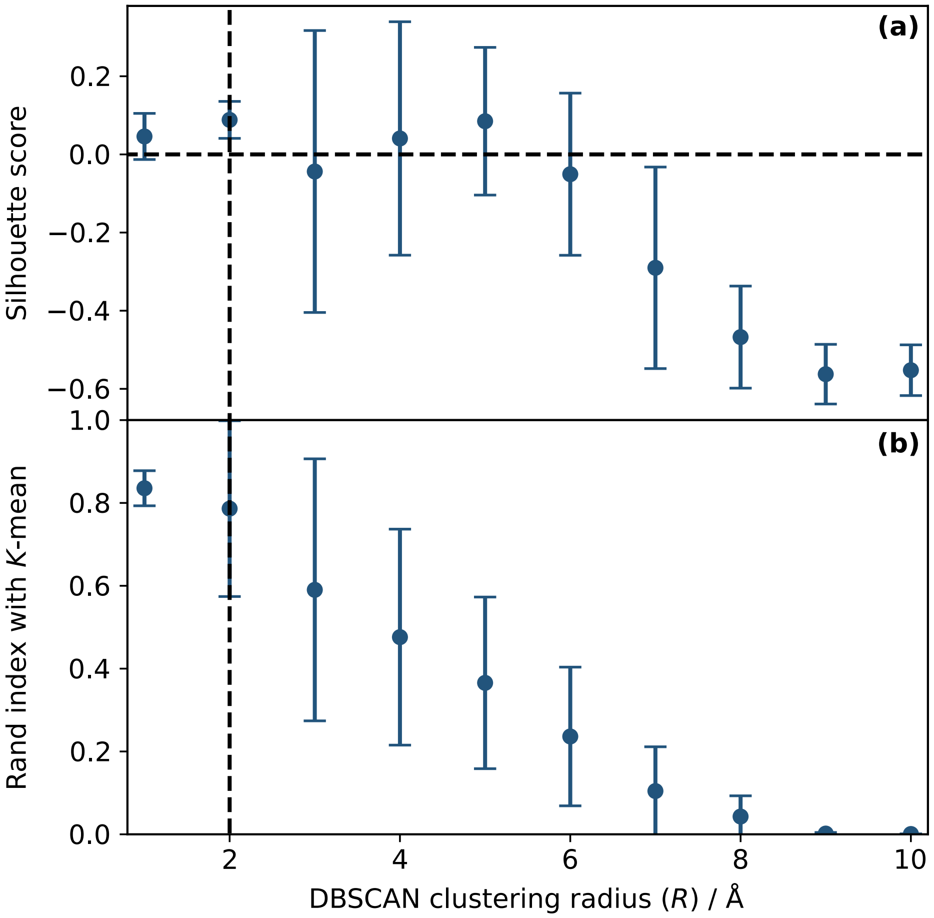

Figure 4.4.14: Identification of the ideal DBSCAN clustering radius, R. We show the variation of R with **(a)** silhouette score, (b) the rand index between clusters found with DBSCAN and a K-means equivalent with K set to the number of clusters given by DBSCAN. The averages given are over the 9 proteins used in this benchmark, with the error bars providing the standard deviation. The vertical dotted line indicates the chosen value of R, the horizontal dotted line the zero-silhouette score, where points should remain above this to demonstrate safe clustering.

Figure 4.4.14(a) demonstrates that an R of 2 is needed to ensure the points are well matched to their clusters, which agrees with the assessment of the rand index in Figure 4.4.14(b). Note that the rand index at large R drops to zero because all of the data was placed in a singular cluster – which we defined as NaN, in actuality this would return 1.0 as the cluster defined by DBSCAN and K-means would match exactly.

Although the silhouette score at $R = 2$ is positive, it is still small at 0.09, demonstrating the poor quality of the data. It is challenging for DBSCAN to distinguish clear clusters, as the search particles remain seemingly randomly scattered following convergence. Furthermore, there is a lack of quaternion convergence at solutions that do have their translational components converge, in contrast to the analytical benchmark in Section 4.4.3.1. Coupled with the decreased number of clusters with increased iterations at large R, we can conclude that the minima of the PES are not well defined. Thus, Equation 4.4.7 is too soft a potential for this exercise. This makes intuitive sense looking at the impact of k on the slope of the energy profile in Figure 4.4.10. In some cases, we find that the ligand can only take a few steps towards the receptor before convergence is achieved, prior to any surface contact. Initialising the docking with the ligand in a surface contact position goes some way to amending this, but manually setting up every protein in such a manner, where there are many possible starting positions for each, is time-consuming and not desirable for a protein docking engine.

4.4.4.3 Protein docking success rate

Taking the clustered results at an $R = 2$, we return a success rate of 0.0% for all cases. However, when we calculate the score of ground truth, we found that in 4 our of 9 cases it places competitively, with three of these corresponding to a score higher than anything found through blind docking. This further reinforces our conclusion that the gradients are too shallow. Therefore, we require a scoring function with more distinct gradients, and an algorithm that both initialises the ligand closer to the receptor, preferably in a surface contact position, and generates solutions faster than the current method. Our work in Section 4.5 focuses on our efforts to create this algorithm.

4.4.5 Discussion

In conjunction with the clear gradients returned through the SDF of our objects, our analytical energy profile works exceptionally well for the toy models, where the binding site is well-defined. Indeed, even in the scenario where there are multiple high-scoring sites available to the ligand sphere, and the target site is occluded, the ligand can navigate through the receptor and find its way into the binding pocket. This demonstrates that the Adam optimiser, coupled with the PES defined by our analytical scoring function, can navigate through or around energy barriers in search of global minima. This would suggest a resolution to the PSO boundary condition of JabberDock, where it was unable to produce solutions that required finding an occluded site. However, when applying this approach to a small benchmark of proteins, we find that, in general, the success rate is lower than the previous iteration of JabberDock. The highest quality model regime occurs around high isovalues (0.67 – 0.79), in contrast to JabberDock. Using isovalues within this range contradicts the general discussion about the physicality of the STID isosurface representation made in Sections 2.3.4 & 2.5. This shift in the isovalue is due to necessary shape intersection; where JabberDock penalised but allowed overlap provided it facilitated better surface complementarity.

This analytical approach does not permit any intersection. Complexes with a significant amount of hydrogen bonding through the water at the interface, therefore, perform well with this analytical method, since the boundary of the isosurface is well defined as a pseudo-excluded volume to water, as discussed in Section 2.5. Complexes featuring more direct intermolecular bonds require these regions of side-chain uncertainty to have some overlap. Increasing the isovalue to ignore this uncertainty is contrary to the idea behind the STID map, and therefore the first aim laid out in Section 1.5; thus, we aimed to retain the pre-defined isovalue ~0.43, and use a semi-empirical solution to permit some level of overlap.

Based on a Lennard-Jones energy profile, our semi-empirical scoring function contained a sliding hyperparameter, k, that permitted some level of intersection depending on its scale. Following a round of meta-optimisation, we determined the best k value that returned the best average score at the known ground truth was 4.16. Taking this value of k and following a similar Adam-based optimisation procedure as the analytical method, returned a success rate of 0.0%. However, when aligning the unbound monomeric units into their bound complex counterparts, we found that these ground truth scores would correspond to high-ranking solutions, and if found natively would push our success rate up to 44.4%. This suggests that the gradients to the minima are ill-defined and that our energy profile is too soft. Utilising a scoring function similar to the analytical benchmark, except with terms accounting for intersection, is, therefore, more likely to yield solutions that represent true minima.

We find that both our approaches perform poorly when we consider solution convergence. On expanding the number of initial search particles, we see an increased number of clustered solutions. This indicates, for both the analytical and semi-empirical PES, that there are isolated lower-energy minima that remain undiscovered. This is despite the evidence from the toy models that the ligand can pass through energy barriers to dock into such minima. While the rotation quaternions in the analytical approach are found to converge at specific translation positions, this was not the case for the empirical method. Increasing the number of search particles, or running additional epochs to find more solutions, is impractical both from a benchmarking and general user perspective. It is unknown if this will return higher-quality models compared with JabberDock, and current results suggest otherwise. Of particular concern, is the time taken for these calculations to be performed. The timings for both benchmarks, for the quality of the returned models, imply that there is little to be gained at significant computational cost, particularly considering a full benchmark on the 230 water-soluble test protein complexes[2] will be required. Resolving this issue requires either a vastly quicker sampling method, or a PES with a lower frequency. Generating a PES with a smaller Lipschitz constant will require an adapted scoring function or a completely new one. With our current SDF approach, an option such as Margin Ranking Loss,[8] or

Warp Loss[9] is attractive, as it allows us to perform meta-optimisation to permit shape intersection, while iteratively funnelling the surrounding space towards the ground truth. Regarding a sampling strategy, in the next section, we shall discuss our novel solution to this problem, with an algorithm capable of rapidly generating solutions based on ray-tracing methods.

4.4.6 Methods

4.4.6.1 Designing the toy models

We begin by defining a spatially fixed receptor sphere, $S_{\mathrm{R}}(\lambda)$, in 3D:

$$S_{\mathrm{R}}(\lambda) = \sqrt{(x_{\mathrm{v}} - x_0)^2 + (y_{\mathrm{v}} - y_0)^2 + (z_{\mathrm{v}} - z_0)^2} - r_{\mathrm{R}} \ ; \ \lambda \in \mathbb{R}^{n \times 3}, \tag{4.4.8}$$

where x_{v}, y_{v} and z_{v} are meshgrid points corresponding to the Cartesian grid; x_0, y_0 and z_0 are coordinates to shift the sphere into different regions of space, and r_{R} is the radius of $S_{\mathrm{R}}(\lambda)$. The number of meshgrid points defines the resolution of the sphere. The ligand sphere will be defined in a similar manner, but will require seeding across the search space. A binding site can then be created by first generating the ligand, $S_{\mathrm{L}}(\lambda)$, in the known binding position(s) with radius r_{L}. Subsequently, we can generate our receptor sphere with a binding site, $S_{\mathrm{R,mod}}(\lambda)$, *via* the following:

$$S_{\mathrm{R,mod}}(\lambda) = \max\{S_{\mathrm{R}}(\lambda), S_{\mathrm{L}}(\lambda)\} \tag{4.4.9}$$

Due to the marching cubes Lewiner algorithm applied to generate the sphere's and binding sites, this modifed receptor sphere SDF does not satisfy the Eikonal equation. We therefore apply a cubic smooth-min function to return a smoothed receptor sphere SDF, $S_{\mathrm{R,mod,Smooth}}(\lambda)$, to remove edge effects:

$$S_{\mathrm{R,mod,Smooth}} = \min\{S_{\mathrm{R}}(\lambda), S_{\mathrm{R,mod}}(\lambda)\} - \frac{1}{2}m(\lambda)^3 \tag{4.4.10}$$

where $m(\lambda)$ is defined as:

$$m(\lambda) = \frac{1}{2}\max\{3 - |S_{\mathrm{R}}(\lambda) - S_{\mathrm{R,mod}}(\lambda)|, 0\}. \tag{4.4.11}$$

When creating the receptor sphere with multiple binding sites, Equation 4.4.9 can be applied iteratively with $S_{\mathrm{L}}(\lambda)$ in different positions representing the possible binding positions. Equation 4.4.10 only needs to be applied at the end with the final modified receptor SDF.

4.4.6.2 Generating points on an ellipsoid

Initialising our starting translations is suboptimal when the receptor is elongated (see Figure 4.4.5), as it can lead to both oversampling of certain regions, and is computationally inefficient. It is therefore preferable to map starting translations from our initialised sphere that surrounds the receptor, to an ellipsoid. However, generating points on an ellipsoid's surface uniformly and at random is not analytically possible, unlike the sphere. Consider some long cigar-like ellipsoid, $a \gg b = c = 1$, where a, b and c lie on the Cartesian axes. The density of points at the cigar's tips ($x \approx \pm a$) will be close to that of a unit sphere, yet at $x \approx 0$ the density decreases as $1/a$ relative to the density of points on the sphere. We can approximately resolve this by generating the points uniformly on a unit sphere first, then mapping these points to the ellipsoid scaled to encompass the receptor, $f : (x, y, z) \mapsto (x' = ax, y' = by, z' = cz)$ before discarding points based on some probability associated with it being close to a tip, $p(x, y, z)$.

We begin by generating our points uniformly and at random on a unit sphere using the marching cubes Lewiner algorithm, which is topologically correct even for a very coarse number of points. We then find the product of our two largest principal axes of the receptor, $a \times b = u_{\text{max}}$, before using this as the basis of some acceptance criteria, a_{xyz}, associated with mapping a point from the unit sphere, $u_{\text{xyz}} = (u_{\text{x}}, u_{\text{y}}, u_{\text{z}})$, onto the ellipsoid surface.

$$a_{\text{xyz}} = \sqrt{(u_{\text{x}} \times b \times c)^2 + (u_{\text{y}} \times a \times c)^2 + (u_{\text{z}} \times a \times b)^2}. \tag{4.4.12}$$

The probability of a successful mapping is therefore given by:

$$p(x, y, z) = \frac{a_{\text{xyz}}}{u_{\text{max}}}. \tag{4.4.13}$$

It is because of this probability that we observe noise in Figure 4.4.5. If we moved to a much larger number of initial sampling points, it is expected that this noise would decrease. If a mapping is accepted, the new point, u'_{xyz}, is then transferred to the ellipsoid surface:

$$u'_{\text{xyz}} = (u_{\text{x}} \times a, u_{\text{y}} \times b, u_{\text{z}} \times c).$$

The exact number of points removed by this procedure depends on the change in surface area between the sphere and ellipsoid. A typical mapping would see the removal of 30% of the initial number of points, which tends to zero as the receptor protein becomes more spherical.

4.4.6.3 The Adam optimiser

The Adam optimiser, as applied through the Geoopt Python package[6] as a subclass to the Adam optimiser in Pytorch,[10] is an extension to stochastic gradient descent. It was used for the all of the work requiring gradient descent. It has two key advantages over traditional stochastic gradient descent:

1. It utilises the momentum of descent to optimise convergence. More specifically, it combines some fraction of the gradient from the previous update with the current, allowing it to reach a minimum faster. This also damps oscillatory behaviour.
2. Each parameter has its own associated learning rate, which are adapted independently. In our case, the parameters are the translations and rotations, and k during the meta-optimisation in Section 4.4.4.1.

At each iterative step of optimisation, the following occurs:

1. The current gradients are set to zero.
2. The ligand SDF is roto-translated according to what the optimiser suggests *via* an affine grid built using that round's parameters. In the empirical method used in Section 4.4.4, the ligand boundary points are moved instead.
3. The loss/score is calculated using an appropriate scoring function and is subsequently backpropagated, *i.e.* the gradients are accumulated.
4. According to these gradients, the parameters are altered in the direction of minimising the loss.
5. The gradients from this round and the previous are used to inform Adam of any necessary changes to the learning rates.

Convergence is achieved once the change in loss between steps equates an arbitrary value of less than 0.001. The starting loss is several orders of magnitude above this.

4.4.6.4 Particle Swarm Optimisation

Particles are initialised following the process described in the first paragraph of Section 4.4.6.7. The search space is defined as a Euclidean 3D translation, and a rotational quaternion restricted to the surface of a unit sphere. The particle swarm optimisation solver was employed through the PyManOpt Python package.[11] PSO is used to explore the PES over 50000 iterations using a population size of N initialised agents – search particles (where $N < 1000$ due to the ellipsoid mapping). A social level of 2.0 is used to facilitate communication between particles about the shape of the PES, with a nostalgia level of 1.4. Practically speaking, these quantities affect the velocity update of a particle at each iteration. A particle's updated position contains contributions from the current inertia, the momentum at the particle's best past position (nostalgia), and the momentum of whichever particle in the population is at the current best position (social). Note that there is no "kick and reseed" element of this procedure (see Section 3.2.1.1), risking particles getting stuck in local minima – hence the larger population size to compensate.

4.4.6.5 Toy model initialisation

Both receptor and ligand are generated with their centres of masses at the origin. Ligands were initalised by translating them onto the surface of a sphere with three times the diameter of the receptor. This sphere was defined *via* a marching cubes Lewiner algorithm, and therefore is physically made up of a polygon mesh of triangles. One particle was placed for every 20 triangles used to define the sphere's surface, resulting in 30 different initialised positions. Each triangle has the surface area of 0.5 a.u.^{2} (Arbitrary Units), which maps to 0.5 Å^2 for protein systems. The size of this initialisation sphere was partly to test strength of the gradients at large distances from the receptor, and partly for visualisation purposes in Figure 4.4.3. While rotations are redundant with the ligand toy model, we still applied a unit quaternion parameter to keep the code consistent between methods.

4.4.6.6 Analytical protein docking initalisation

Both receptor and ligand are generated with their centres of masses at the origin. We begin by measuring the three principal axes of the receptor and ligand, and taking the sum of the largest from receptor and ligand plus some buffer (25 Å) as the diameter of a sphere. Ligand translations are then placed on this sphere at three different sampling granularity levels, one for every 12, 16 and 20 triangles, where each triangle has a surface area of 0.5 Å^2. Particles are then mapped to the ellipsoid surface following the procedure outlined in Section 4.4.6.2. Taking the largest complex (PDB: 3EO1) as the maximum limit to the number of points, we return 840, 474 and 312 initial translations per sampling granularity level on the surface of the ellipsoid. In contrast, the smallest complex (PDB: 1AY7) produced 258, 156 and 90 translations. These translations are placed on the CUDA device as a parameter with an initial learning rate of 0.005. Note that Pytorch's affine grid functionals use an internal coordinate system, so a 0.005 learning rate equates to a change of 0.005 × half the number of SDF points in the system along each axes.

Only two rotations are applied per translation. The first is the unit quaternion, the other is the quaternion required to spin the protein 180° along the axis that is perpendicular to the vector that connecting ligand and receptor centres of mass following the translation. This was to minimise the number of initial search particles by having two individual instances where the side of the ligand SDF facing the receptor is entirely different. This reduction is suitable given the convergence observed for the final quaternions in Section 4.4.3. Thus, the total number of search particles is double the number of initial translations on the surface of the ellipsoid. During optimisation, each quaternion is projected onto a unit sphere to prevent scaling/warping of the ligand SDF. Similar to the translations, a starting learning rate of 0.005 was applied here, equating to change of approximately 0.3° in any direction.

4.4.6.7 Empirical protein docking initalisation

Both receptor and ligand are always initialised with their centres of mass at the origin. For the PSO parameter scan, we scattered 1000 translations across the surface of some sphere with radius 1.5 × the largest axis of the receptor grid. These points are then mapped to the surface of some ellipsoid (Section 4.4.6.2), leaving N search particles for each protein. Each of these search particles was given an associated, random rotation quaternion from a unit sphere. These initial points were then fed into the Particle Swarm Optimisation solver, as described in Section 4.4.6.4. This process was repeated 75 times, each with a novel set of translations and rotations – resulting in $75\times N$ individual starting locations in the search space, although only N particles communicated with one another at a time.

For the meta-optimisation of the overlap hyperparameter k, we scattered 300 ligand positions uniformly on the surface of a sphere around the receptor, before mapping these points onto an ellisoid. Each search particle was then provided with a random rotation quaternion constrained to the surface of a unit sphere. The learning rate for the outer-loop (the RMSD loss to alter k) was initially set to 0.1, the inner-loop (Equation 4.4.7 loss to alter the roto-translation parameters) to 0.01. As a brief benchmark, we did increase the number of initial points to 500 for the 1HE1 and 1AKJ cases, but this yielded little change in k, indicating convergence in the hyperparameter. These initial search particles are then used in the inner-loop Adam optimisation procedure, as described in Section 4.4.6.3, with the results used to inform the outer-loop of any necessary changes to k to improve the score of the ground truth.

In the final production stage, we scattered 200 ligand positions uniformly on the surface of a sphere around the receptor, before mapping again to the surface of an ellipsoid. Each search particle was then provided a random quaternion restricted to the surface of a unit sphere. The learning rate was set again to 0.01. Following a full round of Adam optimisation for the translations and rotations through each loop, we repeated the entire process, including the random initialisation stage, until we had 5 repeats in total to ensure we had a greater number of search particles, on average, than the analytical benchmark. This allowed us to monitor solution convergence with the number of clusters, similar to the analytical method's sampling granularity level check.

4.4.6.8 DBSCAN Clustering

JabberDock utilises a K-means based approach to identify similar solutions and cluster them (see Section 3.2.1.1). However, despite the data being Euclidean (well suited for K-means) it has two key weaknesses:

1. We require a number of clusters parameter, K. The value of 300 used by JabberDock is an arbitrary pick. While this demonstrated success in Chapter 3, it may still not be an optimal pick for our dataset. It also means that noisy or outlier points can be bundled with incorrect clusters.
2. There is no consideration of the magnitude difference between the rotation and translations, *i.e.* the translation quantities, between 0 Å and the size of the receptor, are almost always larger than the axis of rotation, which is between -1 and 1. This is somewhat resolved by using the angle as a parameter, but there is still an unfair weighting against the axis of rotation. This could be resolved by normalising all parameters prior to clustering, but this would weight rotations against translations, where translations are the largest indicator of success.

We, therefore, desire a method which does not require the number of clusters to be specified, where the clusters can be of arbitrary shape, and can correctly assign regions of high sampling granularity to a cluster *versus* low sampling granularity. Density-Based Spatial Clustering of Applications with Noise[7] (DBSCAN) can help address these problems. DBSCAN can handle data of arbitrary shape and density. It assigns each point as either a core, border or outlier point. Each of these labels is determined by two input parameters, the radius of a point in a neighbourhood to cluster additional points (R), and the minimum number of points in a cluster (M). Core points are those with at least M points in the specified radius (given by R). DBSCAN classify border points if their neighbourhood contains less than M points, but they are within R of a core point. Combining the border and core points defines a cluster. Outlier points are those that don't satisfy any of these critera. $M > 1$ disregards noise and outliers, but these points are still valid predictions for our data. They should instead be assigned to a single-point cluster. Therefore, for this work, we set $M = 1$. To avoid the second issue discussed above, we initially identify clusters solely through the translations. R should therefore be between 1 and 10 to represent the distances in Ångströms used to define success by CAPRI. DBSCAN clustering was then not applicable for the analytical benchmark, as all quaternions converged within each translational cluster, and it had little impact on the empirical benchmark due to the nature of the poorly-converged data.

4.5 ShapeTracing: rapid generation of protein docking solutions

Author's note: *the results discussed in this Section were obtained by the lead author of the corresponding publication (see Publications & Manuscripts), Adam Leach. Lucas Rudden performed preliminary work with early iterations of the method, and contributed to data interpretation.*

Some of the key issues found with the attempts to improve JabberDock revolve around the speed at which solutions can be found. Generating solutions is key to adapting parameters or hyperparameters, and quantifying success. However, we can not be confident that our current predictions are representative of the true minima, in both protein docking attempts discussed in Section 4.4. This is due to the greater number of identified distinct clusters with increased solutions, demonstrating that regions of the PES have not been fully discovered. Whether these additional minima are of lower energy, or are closer to the ground truth, is a problem that will be addressed in future by reducing the frequency of the PES, while ensuring distinct gradients – namely through scoring function modifications. Another less optimal solution is to apply a brute force method that rapidly samples the space. In our attempts to employ such a method, we developed a sampling algorithm based on ray-tracing. While not as successful as a standalone method *versus* the aforementioned Adam based approaches, it is novel and has potential in protein docking.

Herein, we present a method for the rapid exploration of the search space associated with matching two three-dimensional surfaces of arbitrary roughness and demonstrate its usage for protein docking. Similar to our previous work in this Chapter, both receptor and ligand are represented implicitly as SDFs (see Section 4.4.1). The points defining the zero-contour surface of the ligand can then be moved along a direction, $\vec{v}$, by the minimum distance between the ligand and the receptor, δ, as with traditional sphere tracing, but with a modification to the bound that allows tracing of arbitrary non-convex shapes (see Figure 4.5.1). Our method features two key contributions. First, it leverages on a novel extension of ray-tracing using spheres,[12] derived from first principles, for detecting collisions between approaching non-convex shapes. Second, it adopts an implicit approach for finding where surface contact area is maximised, shown to be effective in an outer-loop Monte Carlo method.

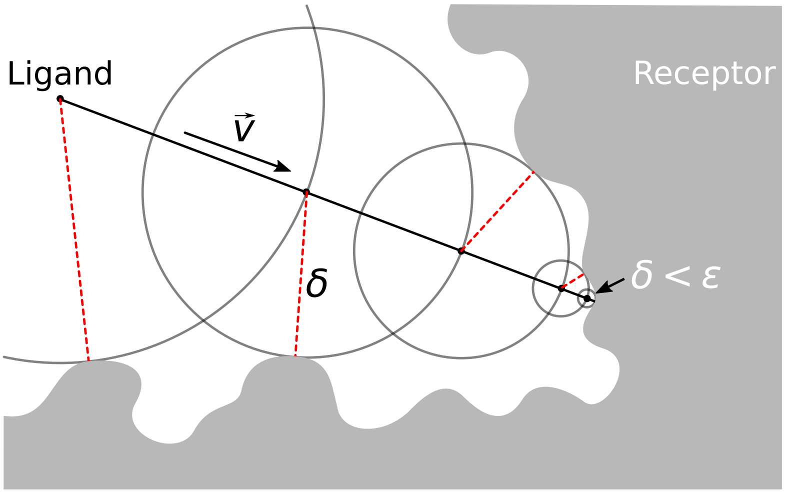

Figure 4.5.1: 2D example of Algorithm 1, steps 3-5 in Figure 4.5.2. The ligand is moved by the closest boundary distance from a vertex on the ligand to the surface of the receptor, δ, along the direction of $\vec{v}$ until within some tolerance ϵ. Here, we scale down the size of ligand for visualisation purposes.

4.5.1 Methodology

The task of identifying the best arrangement of two non-convex protein shapes involves:

1. An inner-loop ShapeTracing algorithm which efficiently moves the shape through space, converging in a few steps, to a bound state.
2. An outer-loop optimiser to initialise the system for the inner-loop, *i.e.* decide on a starting position, and choose a direction for the ligand shape to be fired in the inner-loop ShapeTracing algorithm.

4.5.1.1 ShapeTracing algorithm

The ShapeTracing algorithm updates points, $\lambda_\mathrm{b} \in \mathbb{R}^{n\times 3}$, on the boundary of the ligand SDF along a pre-specified direction $\vec{v}$ until they collide with the receptor. The ligand is moved along the receptor's volumetric grid, with the combined boundary points of ligand with the receptor SDF used to assess whether a collision has occurred in the current iteration, sampled in a manner identical to Section 4.4.4. Specifically, the moving points can be defined as:

$$\lambda' = S(\lambda_\mathrm{b}, \phi_1, l, \vec{v}) \tag{4.5.1}$$

where $l \in \mathbb{N}^3$ is the side length (in voxels) along each axis of the receptor grid. $S(\lambda_\mathrm{b}, \phi_1, l, \vec{v})$ is the ShapeTracing function we outline in Algorithm 1:

1. Compute the analytical intersections from rays cast from the source shape (ligand) at points λ_b in direction $\vec{v}$ to the target (receptor) bounding box (lines 1-2 in Algorithm 1) as in Kay *et al.*[13]

2. If any rays hit (line 4), advance all points λ' by the closest distance (line 5). For glancing rays, push λ' just inside box by the sign of $\vec{v}$ (line 6) as in Willcocks' thesis.[14]

3. Now we know at least one of the points in λ' is inside the bounds of ϕ_1, find the closest distance to the receptor (line 7). Any points sampled outside ϕ_1 are set to ∞.

4. While the shapes are not touching, $\delta > \epsilon$, where ϵ is the tolerance, and while the shape is still inside the bounds of ϕ_1 (line 8), continue moving the whole shape λ' by the closest distance (lines 10-11).

The value for ϵ, 0 by default, can be increased for faster convergence if certain tolerances are acceptable, such as within 1 Å in protein docking. The $\delta/2$ in line 10 reduces the step (by a value proportional to the maximum derivative of ϕ), a common strategy in ray-marching. While in theory we do not need to reduce this step as $||\nabla\phi|| = 1$, in practice $||\nabla\phi|| \approx 1$ due to the discretisation of ϕ. A 2D example of this algorithm is provided in Figure 4.5.1

Algorithm 1 ShapeTracing

```
Input: $\lambda_\mathrm{b}, \phi_1, l, \vec{v}$
Output: $\lambda'$
1  $t_\mathrm{nears} = \max\big(\min\big((1/\vec{v})\cdot(-\lambda_\mathrm{b}), (1/\vec{v})\cdot(l-\lambda_\mathrm{b})\big)\big)$      ▷ Determines whether ray
2  $t_\mathrm{far} \;\; = \min\big(\max\big((1/\vec{v})\cdot(-\lambda_\mathrm{b}), (1/\vec{v})\cdot(l-\lambda_\mathrm{b})\big)\big)$        intersects with receptor
3  intersects = $\{t_\mathrm{nears} > t_\mathrm{fars}\}$      ▷ If true, ray intersects
4  if intersects $\neq \{\}$ then
5      $\delta = \min(t_\mathrm{nears}[\text{intersects}])$      ▷ Distance to move by
6      $\lambda' = \lambda_\mathrm{b} + \delta\vec{v} + \mathrm{sign}(\vec{v})$      ▷ Update coordinates
7      $\delta = \min(\phi_1(\lambda'))$      ▷ Find new minimum distance
8      while $\delta > \epsilon$ and $\delta \neq \infty$      ▷ Continue until shapes touch
9       do
10         $\lambda' = \lambda' + (\delta/2)\vec{v}$
11         $\delta = \min(\phi_1(\lambda'))$
12     end
13 end
```

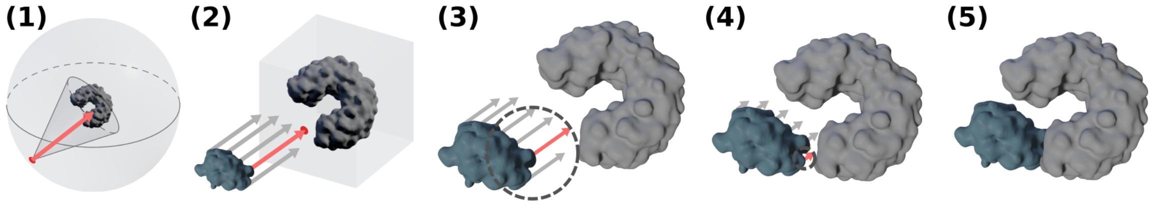

Figure 4.5.2: **(1)** Protein docking with ShapeTracing: the ligand is initialised at random points on a sphere surrounding the receptor with inward-facing cones. **(2)** The ligand boundary points are then analytically moved to be just inside the targets bounding box. **(3-5)** The ShapeTracing algorithm iteratively samples the receptor's signed distance function ϕ_1 at surface positions. The shape is moved by the bound from the closest point (dashed circle, the minimum of these distances).

4.5.1.2 Outer-loop Monte Carlo docking

The ShapeTracing algorithm can be used to generate solutions rapidly. This is demonstrated *via* an outer-loop Monte Carlo docking method, which randomly rotates and moves the ligand to points on a sphere around the receptor, then fires the ligand towards the receptor at a random inward angle in a cone (Figure 4.5.2(1)). This method is outlined in Algorithm 2:

1. Apply a random rotation to the ligand and translate it onto the surface of a sphere surrounding the receptor (lines 6-7, Figure 4.5.2(1)).

2. Set the ray direction, $\vec{v}$ towards the receptor's centre, and apply some random perturbation, γ, within the bounds of a cone (Figure 4.5.2(1), lines 8-9).

3. Fire the ligand at the receptor (ShapeTracing), updating the positions λ' (line 10, Figure 4.5.2(2)).

4. Sum the contact surface area at the solution λ' (Equation 4.5.2), and save the solution parameters if there is more contact than the previous best solution (lines 11-14).

Algorithm 2 Outer-loop Monte Carlo Docking

```
Input: $\lambda_{\text{orig}}, \phi_1, l, \gamma$
$\alpha_{\text{best}} = 0$                                  ▷ surface area to maximize
while true do
    $\vec{t}$ = random point on receptor sphere              ▷ initial translation
    $\vec{c}$ = random point on unit sphere                  ▷ for cone
    $R$ = random rotation matrix                             ▷ for ligand
    $s = \max(l)$                                           ▷ max receptor side length
    $\lambda_{\text{b}} = R\lambda_{\text{orig}} + \vec{t}s$                  ▷ rotate & translate points
    $\vec{v} = (1-\gamma)(-\vec{t}) + \gamma\vec{c}$                        ▷ construct cone
    $\vec{v} = \vec{v}/\|\vec{v}\|$
    $\lambda' = \text{ShapeTracing}(\lambda_{\text{b}}, \phi_1, l, \vec{v})$
    $\alpha_{\text{cur}} = \mathcal{L}(\lambda', k=1)$                       ▷ contact area
    if $\alpha_{\text{cur}} > \alpha_{\text{best}}$ then
        $\alpha_{\text{best}} = \alpha_{\text{cur}}$
        save parameters
    end
end
```

The final score we maximise is the contact surface area between the receptor and the ligand, similar to both Equations 4.4.5 & 4.4.7. This scoring function, much like Equation 4.4.7, can permit intersection with $k > 1$, although in this work we keep $k = 1$.

$$\mathcal{L}(\lambda', k) = \int_{\Omega} \frac{k/\pi}{k^2 + \phi_1(\lambda')^2} \mathrm{d}\lambda' \tag{4.5.2}$$

Ω represents every point within the bounds of the receptor grid. This integral increases as the ligand approaches the receptor boundary and is not influenced by points away from the surface (such as the back of the ligand).

4.5.2 Results

We compared the performance of Monte Carlo with and without ShapeTracing in docking surfaces generated by STID maps. We took the average values of the best docks over 10 runs and, for each run, sampling was terminated after 5000 iterations as further iteration yielded little improvement. Docking using ShapeTracing produced better solutions than a naive sampling of ligand positions with Monte Carlo (see Figure 4.5.3).

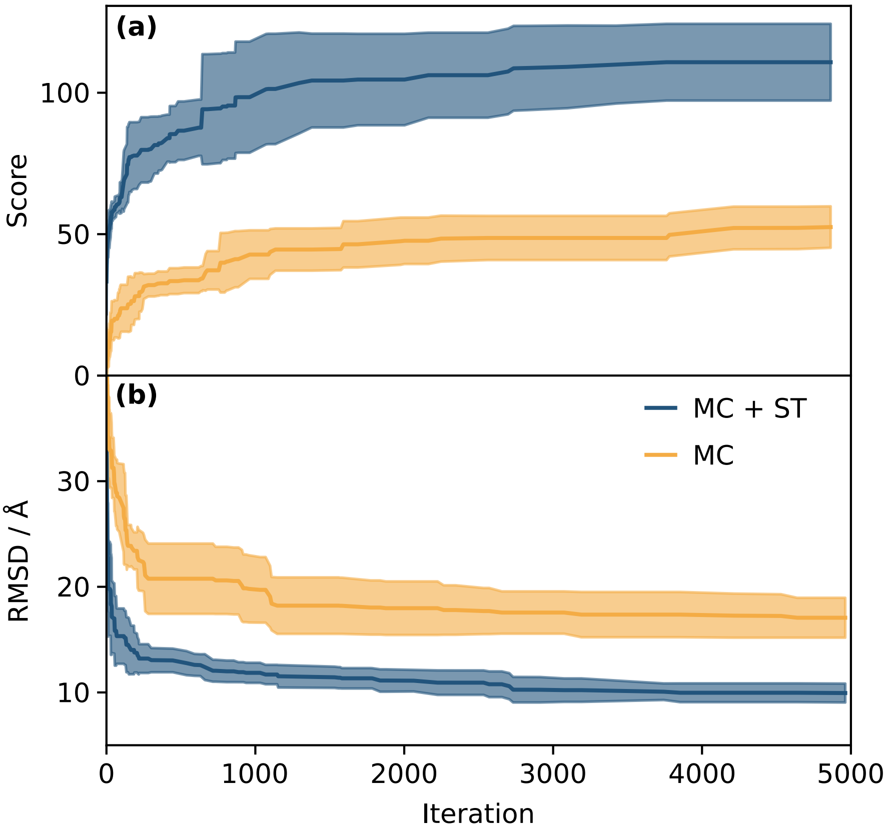

Figure 4.5.3: Combining ShapeTracing (ST) with a Monte Carlo (MC) search improves the performance of STID map based protein docking. MC+ST finds poses with both **(a)** higher score and **(b)** smaller RMSD with respect of the known docked pose. Example results from CAPRI case 1AKJ are shown (unbound ligand 2CLR, unbound receptor 1CD8). Results for 14 additional test cases are available at `https://github.com/cwkx/ShapeTracing`.

We investigated the relationship between cone width and solution quality by varying the cone parameter γ. The RMSD and score after 5000 iterations were averaged over 10 runs. Successful collisions were defined as ligand positions within a tolerence of $\epsilon = 0.0001$ Å from the receptor. We found that, in general, larger values of γ led to worse solutions (see Table 4.5.1), while a value of $\gamma = 0.05$ produced the best results. Slower tracing for higher values of γ is due to near-misses and glancing collisions requiring more iterations than collisions where $\vec{v}$ is perpendicular to the receptor's surface. While Monte Carlo sampling without ShapeTracing is significantly faster, it generally produces solutions with significantly worse scores. In general, ShapeTracing produces more viable solutions over a much shorter time period than our previous efforts in Section 4.4 and Chapter 3.

Table 4.5.1: The impact of the cone parameter, γ, on ShapeTracing (ST), and a performance comparison between Monte Carlo (MC) and MC+ST. Per second refers to result yields from simulations run on one NVIDIA GeForce RTX 2080 Ti. While ST is an order of magnitude slower than MC, it generates four orders of magnitude more docked poses per second.

	MC	MC+ST cone parameter, γ				
		0.00	0.05	0.10	0.15	0.20
Iterations per second	1040	83.8	83.5	82.4	79.5	75.9
Docks per second	0.001	83.8	83.5	82.4	79.5	75.6
Misses per second	–	0.00	0.00	0.012	0.04	0.351
Best RMSD at 5k / Å	17.07	9.94	9.33	9.58	11.0	10.3
RMSD Std dev. at 5k / Å	1.89	0.63	1.10	0.90	0.81	1.35

4.5.3 Discussion

We have found that moving the ligand towards the receptor by the shortest distance to the receptor, and updating this distance at each step, significantly improves the convergence of finding non-intersecting protein docking solutions. The analytical ray-box intersection allows for quick evaluation of far-away solutions. The ShapeTracing algorithm can move arbitrary non-convex shapes with a few, inexpensive operations that can be calculated in parallel on a GPU. Regarding the ShapeTracing method's limitations, the ligand must be initialised away from the receptor, rather than inside it. Therefore, we cannot find occluded solutions without modification. Furthermore, while this approach is sensible for finding exact solutions where contact is maximised without intersection, as we have discussed in length, successful docking poses often feature some shape overlap. Handling such cases requires an energy profile adapted for intersection, either through the overlap hyperparameter k in Equation 4.5.2, or through a scoring function like Equation 4.4.7. However, given that this ShapeTracing method can rapidly generate solutions on a timescale significantly shorter than efforts discussed in Section 4.4, where the computational expense associated with solution generation was the main barrier to resolving objectives 2–5; future work should focus on combining this rapid Monte Carlo + ShapeTracing algorithm with some further inner-loop gradient-based optimisation approach, facilitating rapid and successful protein-protein docking.

4.5.4 Software availability

The algorithm has been implemented using PyTorch on the GPU and is available at `https://github.com/cwkx/ShapeTracing` along with the data for 14 additional test cases.

4.6 Conclusions

In making the case for an updated protein docking algorithm in Section 4.1, we outlined a few key objectives necessary to achieve the desired improvement in success to move beyond JabberDock. In particular, we wanted to:

1. Investigate whether it was possible to harness bound dynamics as a surrogate for another complex with utility unbound docking (objective 1).
2. Develop a means for ligands to find their way into occluded binding pockets (objective 2).
3. Consider a scoring function that scores flat surface contacts well, while still considering the more complex binding interfaces (objective 3).
4. Use a scoring function with ideally no weightings and is analytically derived to promote transferability between different parts of the proteome (objective 4).
5. Speed up the general calculation time and reduce the memory requirements. (objective 5)

These measures would address the key boundary conditions identified in JabberDock, and enhance the utility for everyday users.

We have provided evidence that utilising ligand or receptor dynamics extracted from a bound complex does indeed improve the ability of JabberDock to predict successful models. This indicates transferability at the binding site between monomer units, and therefore future versions of JabberDock should emphasise the use of bound surrogates to improve quality. We have also demonstrated that a few single point mutations intrinsically alters a STID map, thus impacting subsequent docking predictions. This establishes the utility of JabberDock as a tool in mutagenesis work. Future work will attempt to quantify the exact improvement in success rate given the use of these bound complexes, as well as identify which regions of the proteome it is viable for, to determine the benefit for users in different fields.

Converting the STID map at a specific isovalue into a Signed Distance Function allows us to address the remaining objectives. These SDFs, at every point in space, provide information on the distance to the surface – which has significant value in a protein docking scenario. We have shown that an analytically derived scoring function, which is maximised when the surface complementarity is highest, can generate the correct target complex for a toy model regardless of initial position and can correctly predict a solution even when the binding site is occluded. The ability to predict an exact solution regardless of initialisation is a powerful tool in a wide range of computer graphics software, for example in the development of video games. However, in biophysics, an exact scoring function neglects the necessary shape overlap that must occur due to the uncertainty in the side-chain motion, a key feature of our STID maps. Our efforts to address this, lead to the advent of a semi-empirical scoring function with a sliding hyperparameter that allowed for some intersection. However, upon applying this scoring function with a similar optimisation approach, we find that solutions do not converge to known minima (*e.g.* the ground truth), returning models that are not representative of the scoring function's true ranking. In both analytical and empirical cases, we observe that the number of identified clusters does not converge as we increase the number of search particles.

A partial solution to this is a brute force method that rapidly samples the PES to generate preliminary solutions. These solutions could then be considered *via* an improved scoring function which decreases the Lipschitz constant of the search space. In addressing this brute force solution, we developed the ShapeTracing algorithm to sample the PES rapidly. The algorithm is based on ray-marching methods, which efficiently and rapidly moves a shape, in our case a protein-ligand, by the bounds of the distance to some target receptor. At the optimal cone parameter, it was able to generate 83.5 potential docked solutions every second, whereas, on the same hardware, the analytical method in Section 4.4.3 was able to generate 0.03 per second, the semi-empirical in Section 4.4.4 0.01 per second. There is, therefore, several orders of magnitude difference between this ShapeTracing approach and Adam. While the overall results returned by the small ShapeTracing benchmark were less successful, its ability to rapidly generate potential docked solutions makes it an attractive preliminary step in docking. An outer-loop MC+ST method, like that demonstrated in Section 4.5, could quickly provide a set of possible solutions already in a surface contact position to an inner-loop Adam optimisation stage. This inner-loop should then feature a more suitable scoring function that: **(a)** permits minor intersection without using a higher isovalue choice, **(b)** allows the ligand to pass through the receptor in the pursuit of occluded minima. and **(c)** provides clear, well-defined gradients to said minima. While the aforementioned loss functions in Section 4.4.5 could provide some assistance in this regard; given our work in Section 4.4, a scoring function like that of Equation 4.4.5 with appropriate hyperparameters k_1 and k_2 for the Dirac and Heaviside subfunctions could meet this criteria. Furthermore, the successful case of the protein 1FSK, which featured a small binding site (see Section 4.4.3), implies that a scoring function like Equation 4.4.5 will appropriately consider interaction sites which feature minimal surface features from both ligand and receptor. Thus, our ShapeTracing algorithm, in conjunction with an adapted energy profile similar to Equation 4.4.5, should meet the demands of the remaining objectives above, while also retaining the key feature of the STID map; its ability to encapsulate side-chain motion for docking. Further work towards JabberDock 2.0 will, therefore, focus on such an approach, and on addressing the additional objectives we highlighted in Chapters 2 and 3:

1. Building STID maps with non-Gaussian functions modelling the pseudo-electron density, *e.g.* a Lorentzian.
2. Using forcefields other than Amber ff14SB[15] during MD and the subsequent STID map building step, including polarisable models.
3. Adding additional post-processing refinement steps, such as another MD routine.

These remaining areas of interest could further refine the STID map representation and improve our protein-protein docking engine results. In the next Chapter, we will look beyond the impact of local side-chain dynamics on the quaternary structure and function of a protein, and consider larger, domain-level dynamics as we look through the lens of a relatable case study.

4.7 Bibliography

[1] M. F. Lensink, R. Méndez and S. J. Wodak, *Docking and scoring protein complexes: CAPRI 3rd Edition*, *Proteins Struct. Funct. Bioinforma.*, 2007, **69**, 704–718.

[2] T. Vreven, I. H. Moal, A. Vangone, B. G. Pierce, P. L. Kastritis, M. Torchala, R. Chaleil, B. Jiménez-García, P. A. Bates, J. Fernandez-Recio, A. M. J. J. Bonvin and Z. Weng, *Updates to the Integrated Protein-Protein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2*, *J. Mol. Biol.*, 2015, **427**, 3031–3041.

[3] K. . Kim, F. Tao, J. Fuchs, C. Woodward, A. T. Danishefsky, D. Housset and A. Wlodawer, *Creviceforming mutants of bovine pancreatic trypsin inhibitor: Stability changes and new hydrophobic surface*, *Protein Sci.*, 1993, **2**, 588–596.

[4] L. Petrovic, M. Henne and J. Anderson, *Volumetric methods for simulation and rendering of hair*, Pixar animation studios technical report, 2005.

[5] O. Alexandrov, *Wikimedia Commons*, `https://commons.wikimedia.org/wiki/File:Signed{_}distance1.png`.

[6] M. Kochurov, R. Karimov and S. Kozlukov, *Geoopt: Riemannian Optimization in PyTorch*, 2020.

[7] M. Ester, H.-P. Kriegel, J. Sander and X. Xu, Proc. 2nd Int. Conf. Knowl. Discov. Data Min., 1996, pp. 226–231.

[8] A. Bordes, N. Usunier, A. Garcia-Durán, J. Weston and O. Yakhnenko, Adv. Neural Inf. Process. Syst., 2013.

[9] J. Weston, S. Bengio and N. Usunier, IJCAI Int. Jt. Conf. Artif. Intell., 2011, pp. 2764–2770.

[10] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai and S. Chintala, Adv. Neural Inf. Process. Syst. 32, 2019, pp. 8024–8035.

[11] J. Townsend, N. Koep and S. Weichwald, *Pymanopt: A python toolbox for optimization on manifolds using automatic differentiation*, *J. Mach. Learn. Res.*, 2016, **17**, 1–5.

[12] J. C. Hart, *Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces*, *Vis. Comput.*, 1996, **12**, 527–545.

[13] T. L. Kay and J. T. Kajiya, *Ray tracing complex scenes*, *ACM SIGGRAPH Comput. Graph.*, 1986, **20**, 269–278.

[14] C. G. Willcocks, *Sparse volumetric deformation*, *Ph.D. thesis*, Durham University, 2013.

[15] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, *ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB*, *J. Chem. Theory Comput.*, 2015, **11**, 3696–3713.

5 | Dynamics of the SARS-CoV-2 spike glycoprotein

5.1 Introduction

Throughout this work, we have emphasised the importance of accomodating side-chain dynamics when modelling biomolecular systems. In Chapter 2, we discussed its relevance in modelling for a variety of problems. In Chapters 3 and 4, we have discussed its impact in enhancing the quality of solutions for protein-protein docking problems. However, up to this point, we have largely ignored the impact of larger, domain-level restructuring and flexibility, which heavily influences the function of protein complexes. Here, we shall explore the role of tertiary structure dynamics and exploit them to derive conclusions about possible protein-protein interactions, through a relatable case study that has impacted almost every person worldwide.

Since the turn of the millennium, beta coronaviruses (CoV) have caused three zoonotic outbreaks in the global human population. In 2002-2003, severe acute respiratory syndrome coronavirus (SARS-CoV-1) jumped from bat and palm civet groups to humans in China,[1,2] resulting in the infection of over 8000 people and nearly 800 deaths.[3] In 2012, Middle East respiratory syndrome coronavirus (MERS-CoV) crossed from dromedary camel populations in Saudia Arabia,[4] and remains a potential endemic threat to this day in the region. There have been an estimated 2500 cases, resulting in 860 deaths.[5] In November 2019, it was discovered that another SARS-CoV (SARS-CoV-2) virus emerged in human populations in Wuhan, China.[6–8] Phylogenetical analysis suggests that the virus came from local bat species.[9] As of this thesis' submission date (16/02/2021), SARS-CoV-2 has resulted in over 109 million confirmed cases, many of these with long term health issues, and been responsible for the deaths of over 2.4 million people worldwide[10] – the worst pandemic since the ongoing AIDS/HIV pandemic that emerged in the 1970s.

Coronaviruses are visually distinct from other viruses due to their use of a spike glycoprotein (resembling a crown, hence *corona*), to attack the human angiotensinconverting enzyme 2 (ACE2) receptor[9] (see Figure 5.1.1(right)), a transmembrane protein expressed on the surface of cells in the lungs. The spike glycoprotein of SARS-CoV-2, which was quickly resolved following the initial, rapid spread of the virus in Wuhan, is a large homotrimer, each monomer having a mass of approximately 180 kDa (see Figure 5.1.1(left)). It is made up of two subunits, which mediate attachment and membrane fusion.

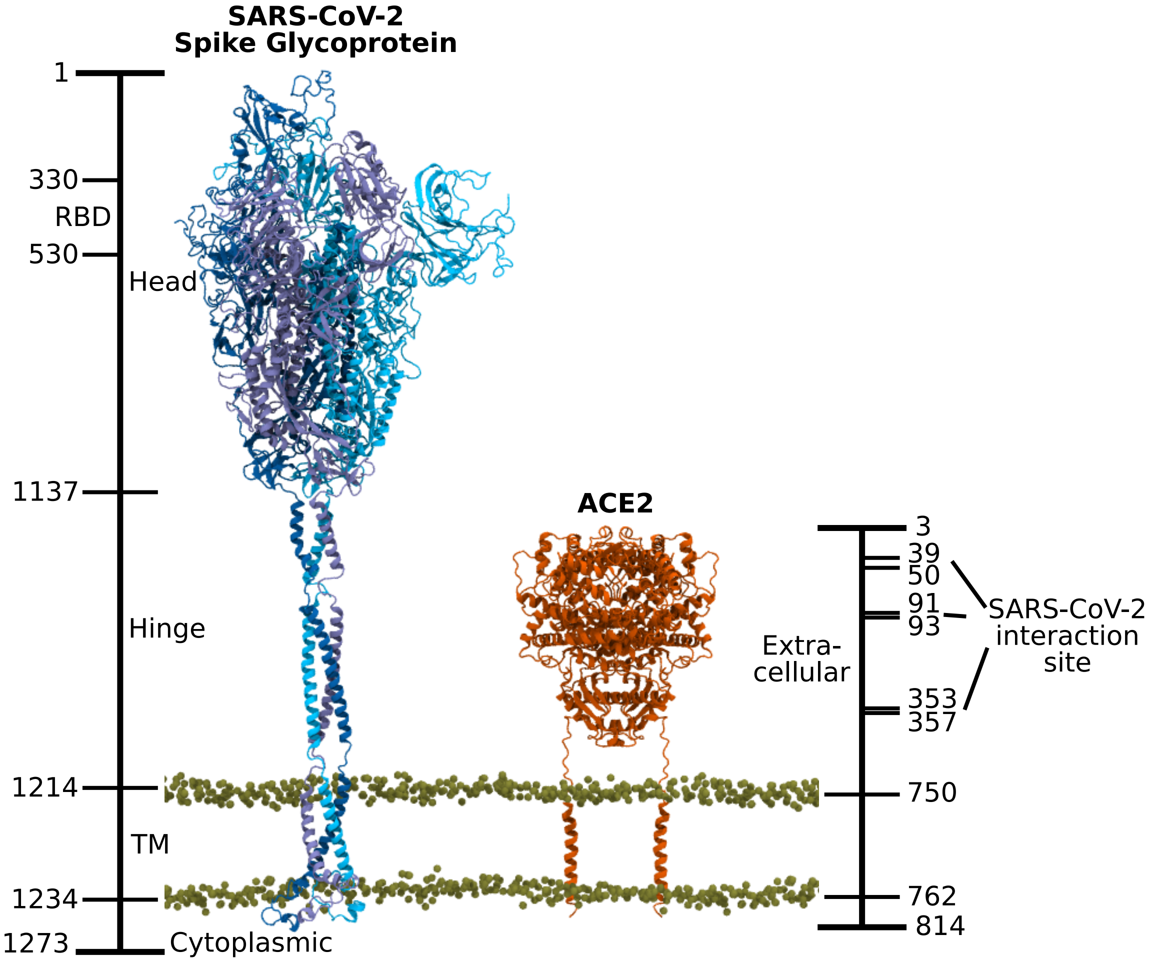

Figure 5.1.1: (left) Full length SARS-CoV-2 spike glycoprotein trimer with one Receptor Binding Domain (RBD) in the open state. Each monmeric chain is coloured in different shades of blue. Residues relevant to our work are indicated with their relative location and domain in the sequence on the left. The membrane position is indicated by the beige phosphorus atoms. Note that the glycans have been removed. **(right)** ACE2 dimer with important residues indicated on the right.

The spike, once in the open state, can use a receptor-binding domain (RBD) to bind with ACE2, confirmed *via* a structure of the SARS-CoV-2 spike RBD bound to the soluble domain of monomeric ACE2 (PDB: 6M0J), which was released in the early stages of the COVID-19 pandemic. In the closed state, the RBD is shielded by the head's surrounding machinery and is therefore inaccessible.[11] The spike is the main target for various antibody therapies and treatments, as preventing the RBD from interacting with ACE2 obstructs viral entry into the cell. Understanding and predicting the potential spike-ACE2 complexes that can form is pivotal to further drug discovery.

It was recently suggested that two spikes could bind to a single ACE2[12] (see Figure

5.1.2(b.i)), and cryo-EM imaging indicated that the spike glycoprotein could cross-link to itself on the virion surface (Figure Figure 5.1.2(b.iii)), leading to a large number of oligomeric spike-ACE2 arrangements.[13] This plethora of complexes would expedite the entry of the virus into the cell, thereby explaining the potency of SARS-CoV-2 over the other beta coronaviruses. We can apply computational methods to assess the likelihood of these different structures to enhance future medical research.

Hereafter, we leverage on six replicas totalling a 4.2 µs-long MD simulation from the Amaro research group of the entire, glycosylated SARS-CoV-2 spike protein in the open state (PDB: 6VSB) simulated attached to a lipid membrane.[14] The hinge/stalk and transmembrane domains (residues 1137-1234) were constructed using homology modelling *via* Modeller,[15] the cytoplasmic (residues 1235-1273) through I-TASSER.[16] Their simulation showed that hinge motion of the stalk region and the presence of the glycans gave the crown a much greater range of motion with respect to the plane of the membrane. Therefore, we used this exhibited flexibility to derive possible virion curvatures that, coupled with the spike's various conformational states, could lead to the dual binding of the spike to ACE2. These results corroborate known virion curvatures and confirm that two spikes can adopt a variety of conformations that invite dual binding events. Indeed, we show that the spike dynamics can even provide information on the necessary macro-positioning of the virion with respect to the prey cell. We also show that we can utilise structural information of the SARS-CoV-1 bound state to inform us on the possible interactions of SARS-CoV-2 with ACE2, generating complexes that agree with recent Mass Photometry data for both cross-linked and single spikes with ACE2 oligomers. Schematics of all possible arrangements proposed through Mass Photometry that we will consider are shown in Figure 5.1.2(b).

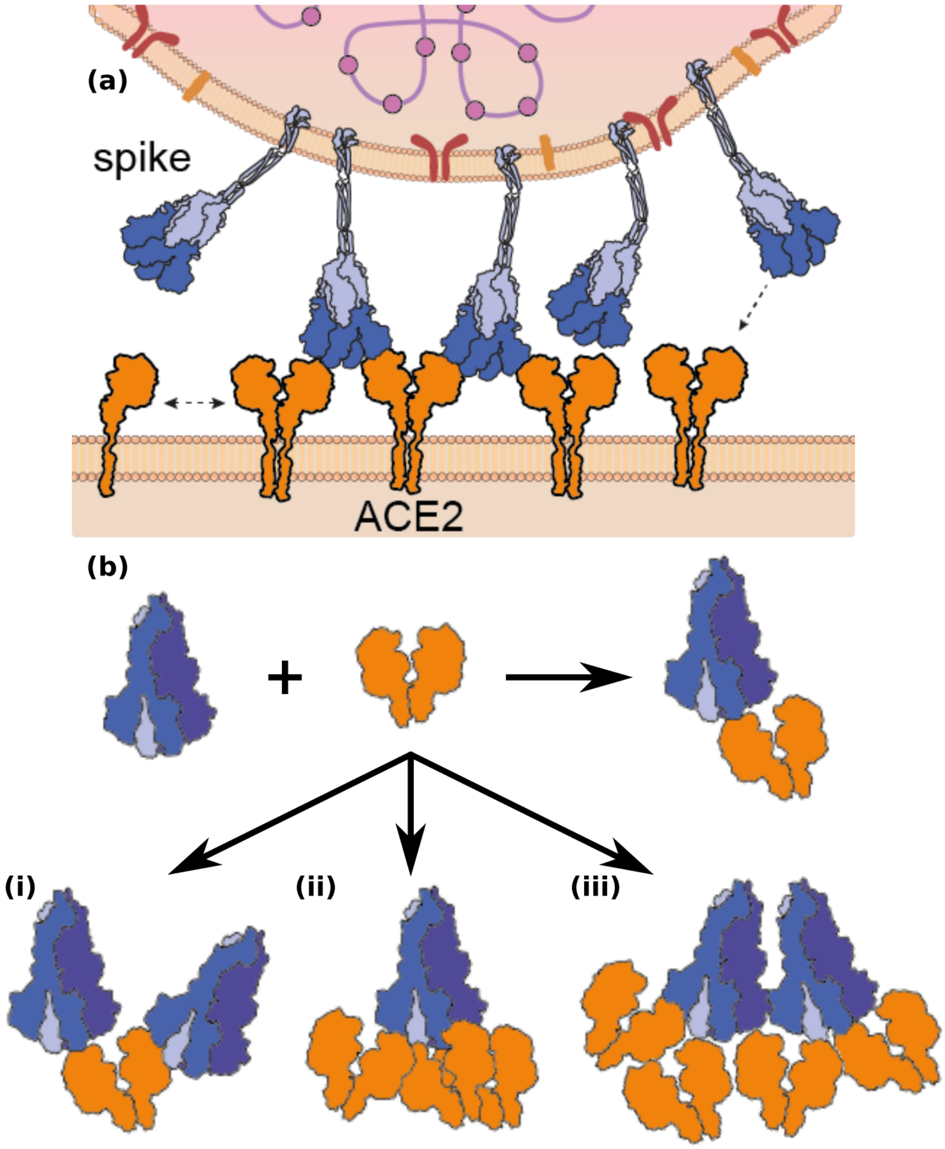

Figure 5.1.2: **(a)** Schematic of SARS-CoV-2 virion and spike glycoprotein (blue) interacting with human ACE2 (orange) anchored to the cell membrane. **(b)** Possible oligomeric spike-ACE2 complexes to be considered in this Chapter: **(i)** 1 ACE2 with 2 spikes (Sections 5.2.1 & 5.2.2), **(ii)** 3 ACE2 bound to a single spike (Section 5.2.3), **(iii)** 4 ACE2 bound to a cross-linked spike dimer (Section 5.2.3). Image adapted from our co-authored manuscript with Olerinyova *et al.* (see Publications & Manuscripts).

5.2 Results

5.2.1 Dual binding mode of the spike is compatible with viral membrane curvature

The binding of two spike glycoproteins onto a single ACE2 receptor is dependent on three main aspects: the conformation of the spike glycoprotein, the position of the virion over the ACE2 receptor and the local shape of the viral membrane (see graphical representation in Figure 5.2.1). We need to account for all of these factors when investigating whether two spikes from the same virion can bind to the same ACE2. For this, we overlay spikes from the Amaro lab simulation,[14] and, from their relative arrangement, we can deduce the shape of the viral membrane they are attached to. We can claim that dual binding is possible if; this alignment occurs without clashes, their associated membrane curvature is plausible, the distance between the spikes expressed on the viral membrane match known distances, and the orientation of the virion relative to the ACE2 cell is within reasonable expectations. We describe the full details for our methods in Section 5.4.1.

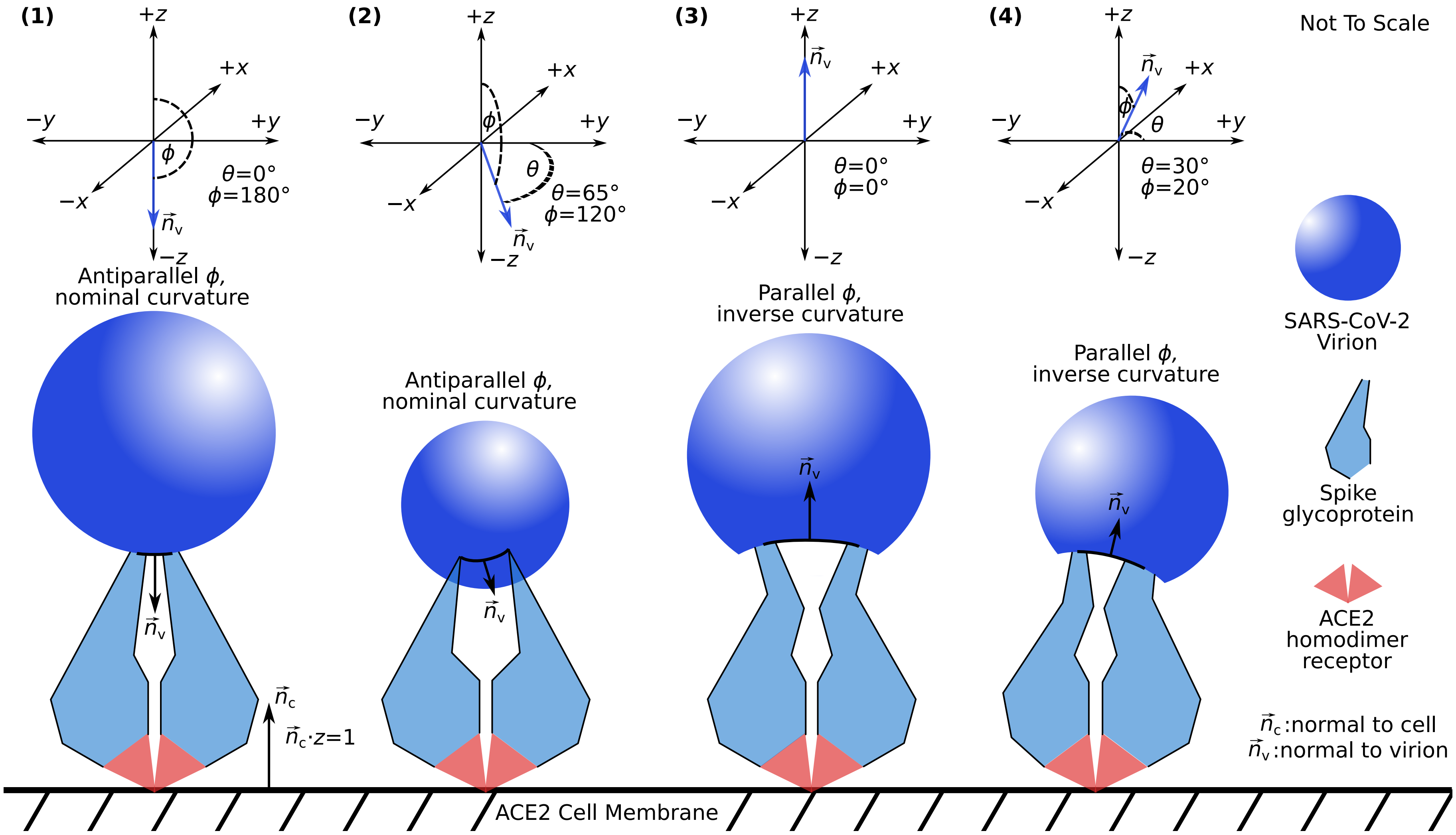

Figure 5.2.1: Schematic for how the curvature of the viral membrane and the orientations of the virion relative to the ACE2 cell membrane are calculated. $\vec{n}_\mathrm{v}$ is defined as normal to the midpoint on the predicted viral membrane between the transmembrane domains. The elevation angle is ϕ, the azimuth θ. The (anti)parallel states here are not relevant for Section 5.2.1, but are defined in Section 5.2.2. **(1)** Antiparallel ϕ state with the virion directly above the ACE2 cell. **(2)** Antiparallel ϕ state with the virion going into the page, normal coming out. **(3)** Parallel ϕ state directly above the ACE2 cell requiring some local deformations in the virion bilayer. **(4)** Parallel ϕ state with the virion going into the page and is twisted along the azimuth.

To generate a template representing two spike proteins bound to the same ACE2, we leveraged on a crystal structure of the SARS-CoV-2 spike RBD bound to the soluble domain of monomeric ACE2 (PDB: 6M0J). We duplicated and aligned 6M0J *via* its ACE2 atoms to each of the chains in the full ACE2 homodimer (PDB: 6M18), obtaining a complete ACE2 receptor with two SARS-CoV-2 spike RBD domains bound. We then aligned pairs of the MD-generated conformations such that their open RBD binds to one subunit of ACE2. For all available protein conformation, models could be built without clashes between spikes (see Figure 5.2.4 in Section 5.2.2). Therefore, for each simulation frame, we calculated the viral membrane curvature associated with the relative arrangement of spikes protruding from their receptor. The methodology details for this are provided in Section 5.4.1. The evolution of the viral membrane curvature over all six simulations, and the maximum and minimum permissible distances between the transmembrane domains is shown in Figure 5.2.2. Overall we find an average nominal curvature of 0.033(17) nm^{-1}, a maximum of 0.099 nm^{-1}, and a minimum of 0 nm^{-1}, where the two transmembrane domains are co-planar. The average inverse curvature (see Figure 5.2.1) is $-0.035(17)$ nm^{-1}, with a maximum of 0.0 nm^{-1} and minimum of -0.074 nm^{-1}. The reported SARS-Cov-2 radii, assuming a spherical shape, are 50–200 nm.[17] In Figure 5.2.2, the corresponding curvature is highlighted with a red band. Thus, 14.3% of the simulation time is spent in conformations corresponding to the nominal shape of the virus.[17] The geodesic distances between spikes (*i.e.* along the membrane connecting them) ranged from 16.2 nm to 58.9 nm, with an average of 33(7) nm. Thus, possible distances between transmembrane domains expressed on the virion surface can range significantly, between 16.2 nm and 58.9 nm, and still allow for dual binding to the ACE2 receptor. This is consistent with distances measured *via* tomograms (10 nm and 80 nm).[18]

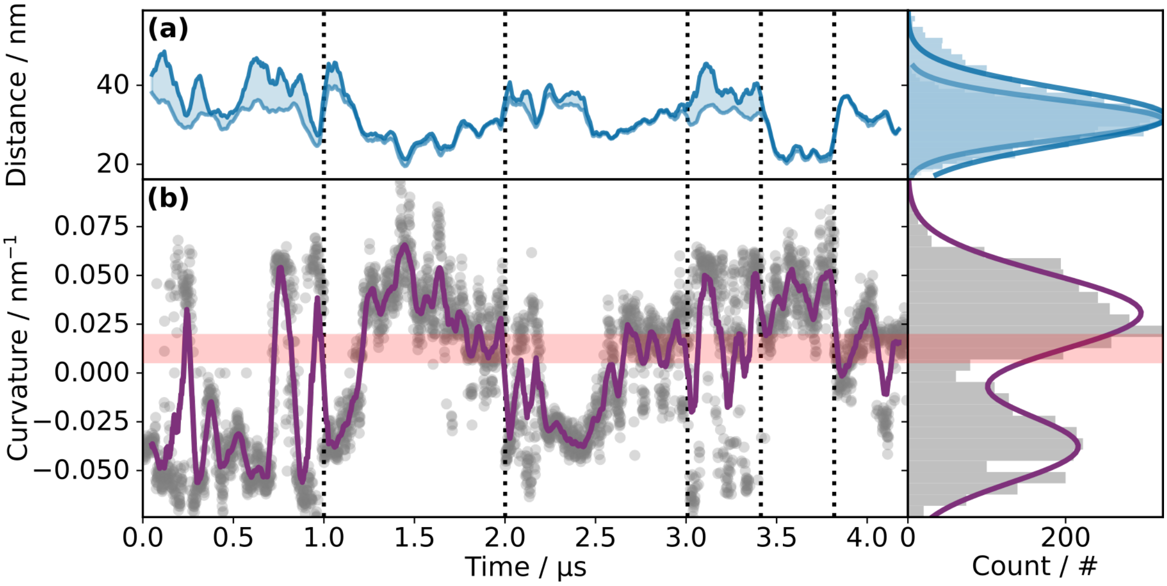

Figure 5.2.2: Evolution in time of: **(a)** Moving average distance between spike protein transmembrane domains. The maximum geodesic distance, or arc length, is in dark blue, the minimum/point-to-point distance in light blue. **(b)** Curvature, with the moving average denoted in palatinate. Each replica simulation is separated by a dotted line. The red band indicates simulated curvatures corresponding to the reported virion radii of 50–200 nm,[17] assuming a spherical shape. On the right, histograms of each quantity have been fitted with Gaussians.

The results described above are obtained by approximating that the two proteins bound to the ACE2 receptor are in the same conformation (as derived from the same MD simulation snapshot). We therefore also analysed the expected viral membrane curvature when the conformations of one protein are derived from MD replica 2, and the other from replica 3, both 1 µs long. Figure 5.2.3 shows the variation in curvature with the azimuth angle (see Figure 5.2.1 for visual definition of the azimuth) for the simulation made up of two different replica of the spike glycoprotein. The red region corresponds to curvatures displayed experimentally by the virus.[17] Large azimuth angles are required to achieve high curvatures, and thus, the membrane at the transmembrane domains sites must deform significantly to accommodate dual binding at these conformations. Therefore, results are comparable to above, indicating that the approximation made by using two identical simulations for our two spike glycoproteins is reasonable.

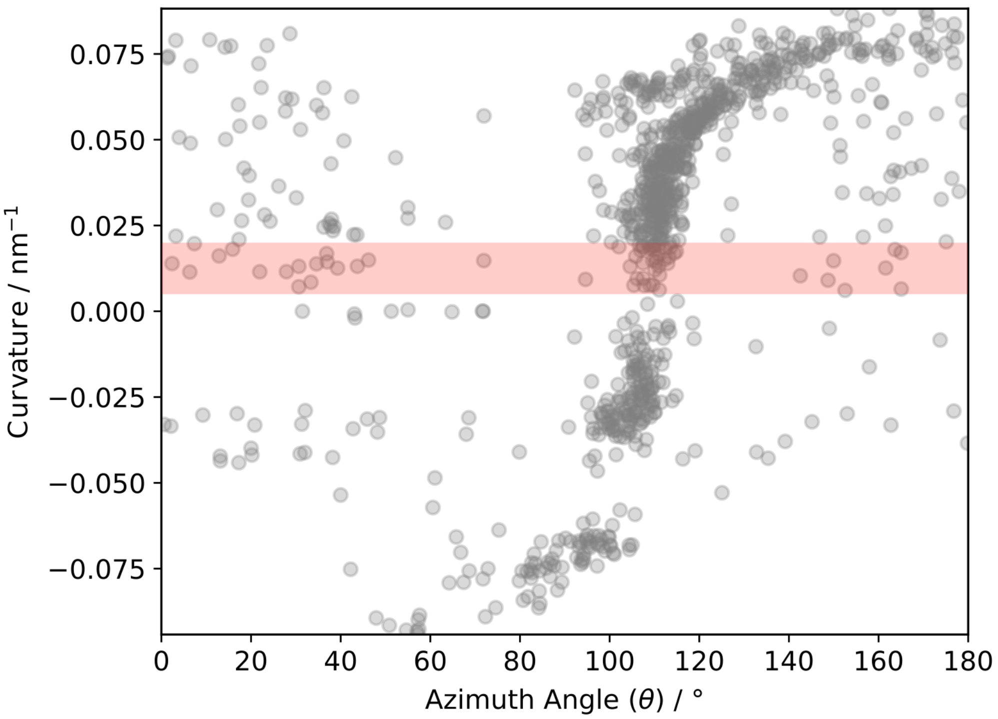

Figure 5.2.3: Relationship between membrane local curvature and azimuth describing the orientation of SARS-CoV-2 virion with respect to the ACE2 membrane (in case of dual binding mode involving proteins in different conformations extracted from MD replicas 2 and 3). The red band indicates experimental SARS-CoV-2 virion curvatures.[17] Extreme rotations with respect to ACE2 are only possible under extreme viral membrane curvatures.

5.2.2 Dual binding mode of the spike is possible under a range of virion-ACE2 receptor arrangements

We analysed possible arrangements of the virion with respect to the target cell's membrane, as derived from the arrangement of spike glycoproteins bound to the ACE2 receptor (see Figure 5.2.1 for visual examples). This position is measured *via* the spherical angles, elevation (ϕ) and azimuth (θ), calculated between the viral membrane normal at the mean position of the two spike transmembrane domains, and the axes defining the position of the ACE2 cell (using the ACE2 cell membrane as a reference frame). States where the viral membrane normal is parallel to the positive z-axis (perpendicular to the ACE2 cell normal) are parallel states ($\phi < 90°$), while the opposite are antiparallel states ($180° > \phi > 90°$). See Figures 5.2.1, 5.2.4, and 5.2.6 for a visual demonstration, and Section 5.4.1 for more details.

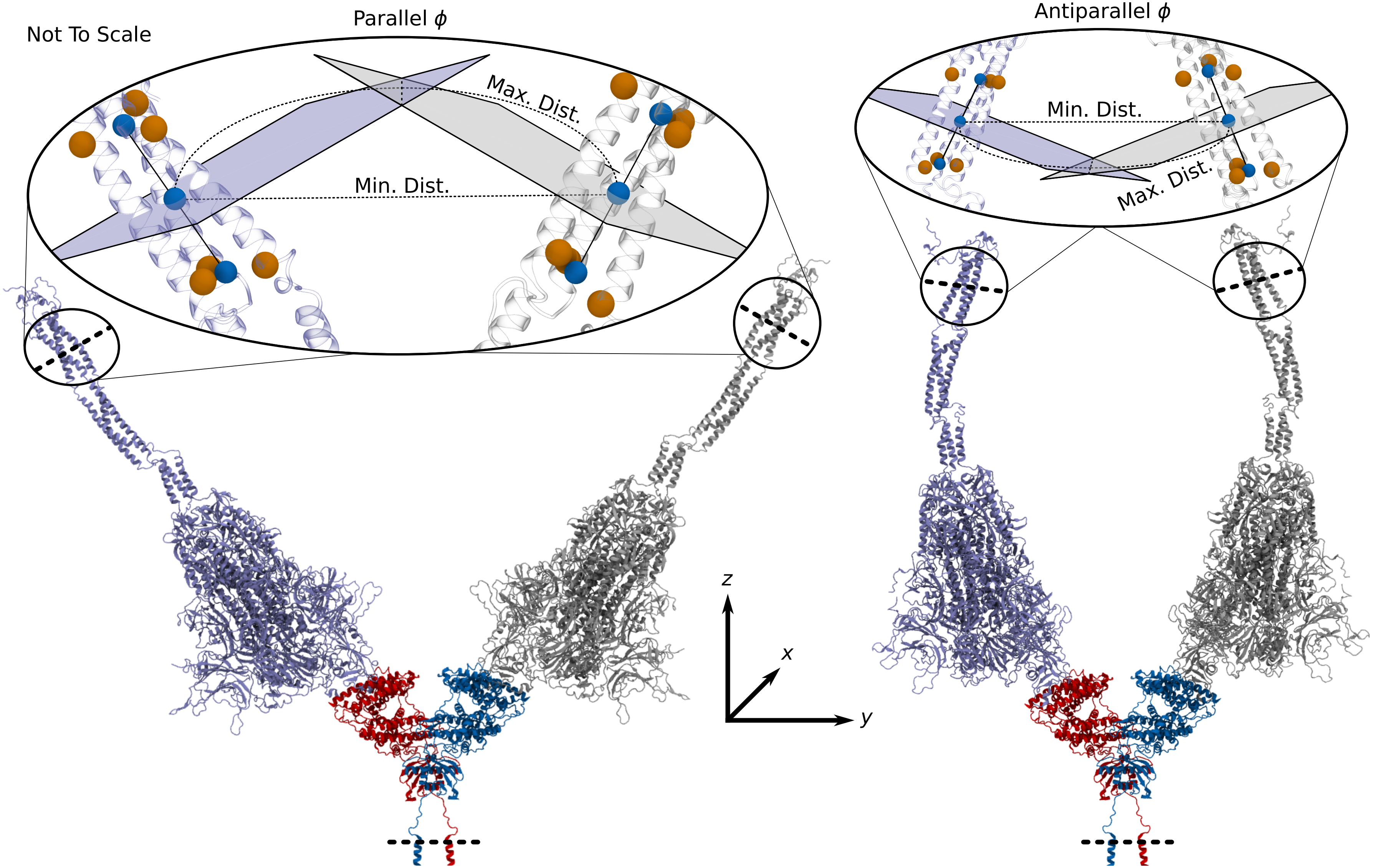

Figure 5.2.4: Proposed structure of two SARS-CoV-2 glycoproteins (violet and grey) bound to the homodimer ACE2 receptor (red and blue). The thick black dotted lines indicates the position of the lipid membrane. The zoom highlights the transmembrane region of the spike glycoprotein, where the membrane-entry and -exit Cαs on residues 1214 and 1234 for each chain are coloured orange. The blue pseudo-atoms indicate the mean position of each residue set (top and bottom), and average of all (middle). The axis that connects all three is defined as the normal to the plane of the bilayer (albeit with a small rotational correction to account for simulation drift), which bisects at the middle blue pseudo-atom. The intersection of the two planes defines the maximum curvature of the membrane, assuming a perfect circular curvature, while the point-to-point distance between the middle blue pseudo-atoms considers the case where the curvature tends to zero. Cases where the z-component of the local viral membrane is aligned with the negative z-axis, or $-\vec{n}_{\mathrm{c}}$, is considered an antiparallel ϕ state. When it is it is aligned along the positive z-axis, it is in the parallel ϕ. See Figure 5.2.1 for a schematic of this concept: **(left)** Conformation in a parallel ϕ state. **(right)** Conformation in a antiparallel ϕ state.

These angles and curvatures are calculated from the simulation made up of replicas 2 and 3, and are therefore more representative of a realistic system. Figure 5.2.5 shows that, in the majority of cases, for any rotation along the azimuth or elevation angle to occur, the alternative angle must be $\approx 90°$. Figure 5.2.6 gives some representative examples of the various more extreme angle states as a visual example. Conformations where ϕ is either ~0° (Figure 5.2.6 (2)) or ~180° (Figure 5.2.6 (3)), are those where the virion membrane inner to outer-leaflet direction is parallel or antiparallel to the ACE2 cell membrane normal respectively. In these states, θ can take any value. In the case where $\phi = 90°$, the spike glycoprotein hinges must bend such that the two bilayers are orthogonal (Figure 5.2.6 (1, 4, 5)). This would likely require some significant deformation of the local virion membrane. States where the head of the spike is at a 90° angle to the virion membrane, while rare, have been observed experimentally through Cryo-EM.[18] The $\phi = 90°$ and $\theta = 90°$ states are therefore feasible given the flexibility of the hinge, but the other extreme angle states, where $\theta = 0°$ (Figure 5.2.6 (4)) or 180° (Figure 5.2.6 (5)) are likely artefacts of deriving virion morphology from the dynamics of the spike glycoprotein. These states would likely result in the membrane sterically clashing with one of the spike glycoproteins, and they would require some significant twisting action of the membrane to accommodate both TM regions.

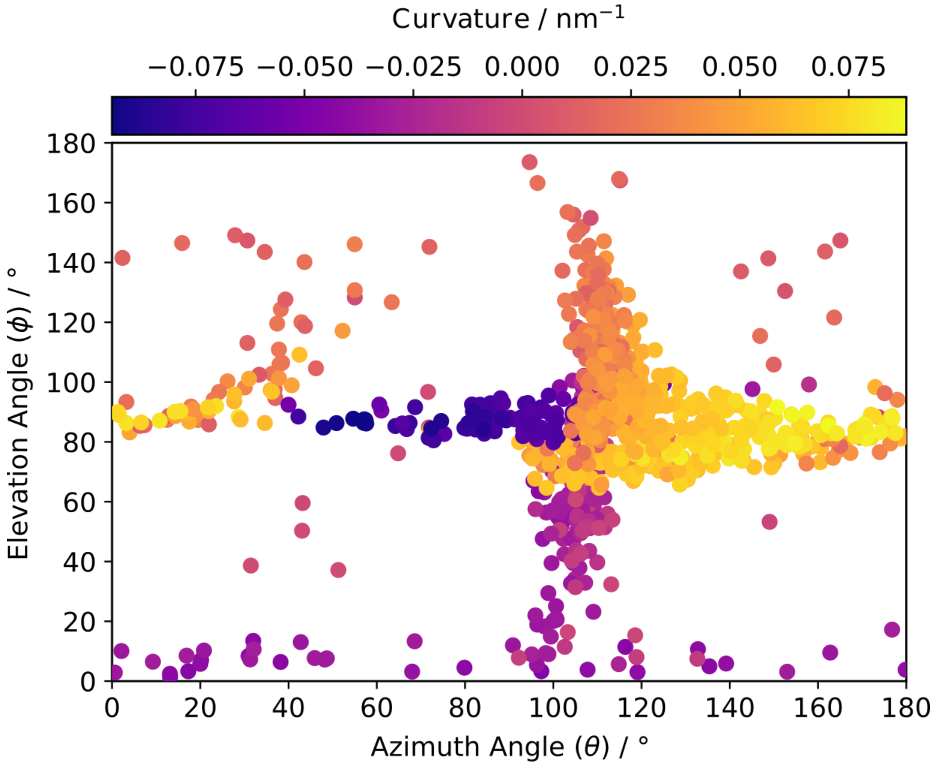

Figure 5.2.5: Predicted orientation of the virion (in spherical coordinates) with respect to the ACE2 membrane. The results were derived from dual binding involving spike proteins in different conformations extracted from MD simulation replicas 2 and 3. Points are coloured according to their associated membrane local curvature. Low membrane curvatures (consistent with experimental data) enable binding according to a range of elevations and well-defined azimuth. Variations on azimuth are possible under extreme membrane curvatures (see also Figure 5.2.3).

Changes in the elevation angle tend to correspond to lower virion's membrane curvatures, and in general antiparallel states correspond to curvatures demonstrated by the SARS-CoV-2 virus.[17] There is also a larger number of nominal curvature (convex membrane) states relative to those with an inverse curvature. These curvatures usually correspond to an antiparallel state, which makes sense considering that a convex membrane in a parallel state would lead to the virus physically clashing with the ACE2 cell. Those few convex parallel states (curvature > 0 nm^{-1}, $\phi > 90°$), and the concave viral membranes (curvature < 0 nm^{-1}), require some form of local deformation in the topography of the membrane. Large curvatures, both nominal and inverse, result from the azimuth angle close to either 0° or 180° (see Figure 5.2.3). As shown in Figure 5.2.6, this would impose an unusual position for the virial membrane relative to the spike TM positions, and are therefore likely artefacts of the MD simulations, though could be facilitated by the formation of blebs or other significant deformations seen through negative stain imaging.[19]

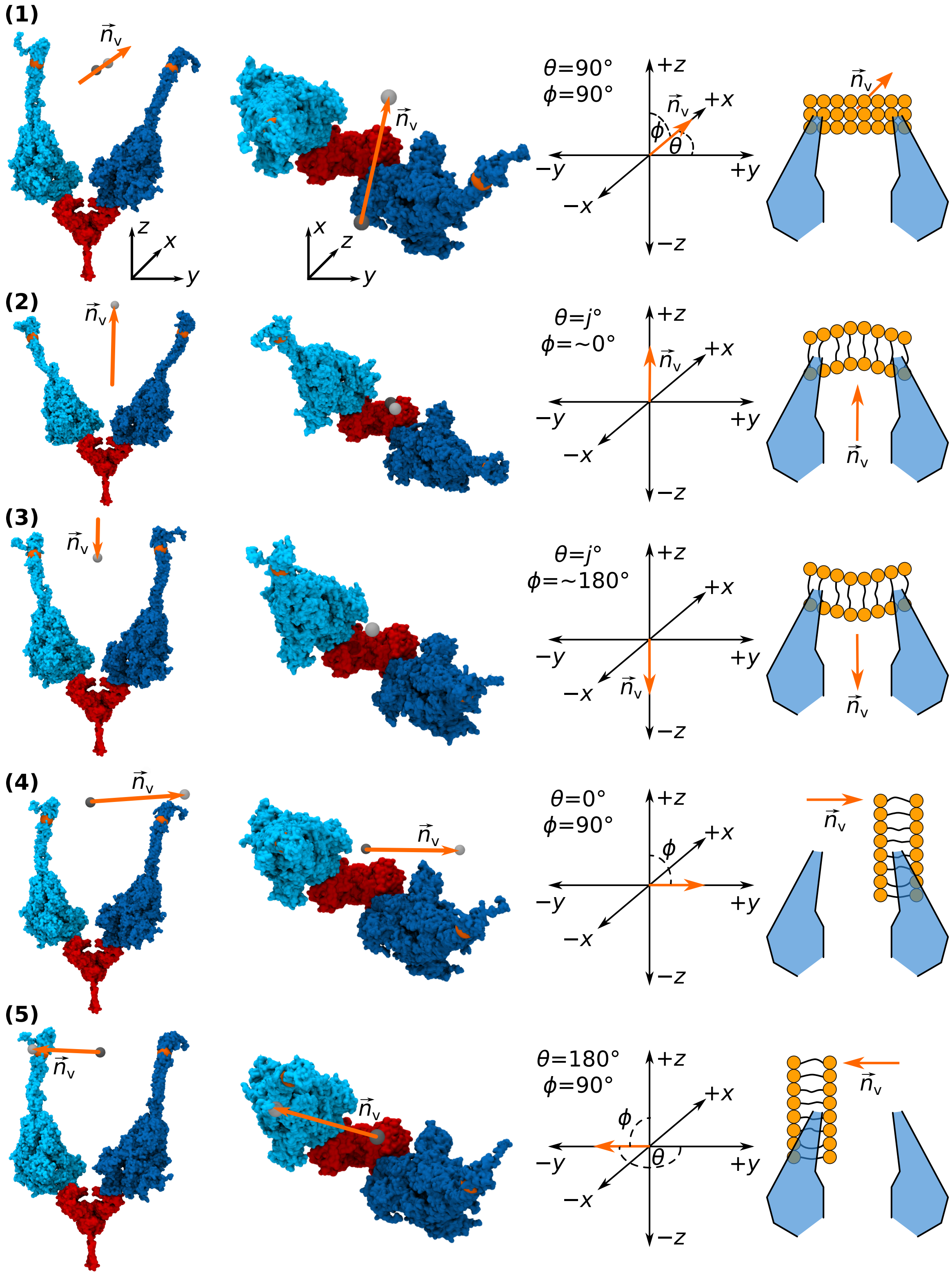

Figure 5.2.6: Extreme angle states from the artificial simulation of replicas 2 and 3. We show a side-on view of the states on the far left, corresponding to the axis seen on the right, and a top-down view for clarity. The red protein is the ACE2 receptor, while the spike glycoproteins are in blue. The light grey sphere represents the intersection point of the two planes that bisect the spike glycoprotein TM regions and is closest to the TM regions (see Section 5.4.1). The dark grey sphere, where visible, represents the centre of the circle from which we model the local curvature. The orange arrow, therefore, is normal to the local virion membrane. We show a schematic of the state on the far right, with the blue shapes representing the spikes and the orange spheres the lipid heads. The azimuth angle (θ) is measured relative to the positive y-axis, the elevation angle (ϕ) to the positive z. j° represents any angle between 0° and 180°.

Examining these angles and curvature for the entire set of replica simulations, we see a similar trend. Figure 5.2.7(a) shows the variation in ϕ, θ and the curvature for all six sets of replica simulations. The variation in curvature with these angles is similar to that displayed in Figure 5.2.5; however, there are two key features in which they differ:

1. In Figure 5.2.7, convex membranes are disallowed at $\phi < 90°$ and concave at $\phi > 90°$.
2. There are some clear structured lines around $\phi \approx 90$ in Figure 5.2.7, whereas in Figure 5.2.5 these are more random.

Both of these differences are due to mathematical artefacts that arise from using two mirror-image simulations. Figure 5.2.7(b) shows the population of states consistent with expectations. $\theta \approx 90°$ states are the most densely populated, where there is essentially no rotation about the z-axis. With the elevation, states with $\phi \approx 180°$ are the most populous, which corresponds to the virus being directly above the ACE2 cell, and, from Figure 5.2.7(a), displaying a convex membrane (Figure 5.2.1(1)). Therefore, we expect this to be the preferred angle of attack for the virion, but we note that other angles are possible given some local viral membrane deformation.

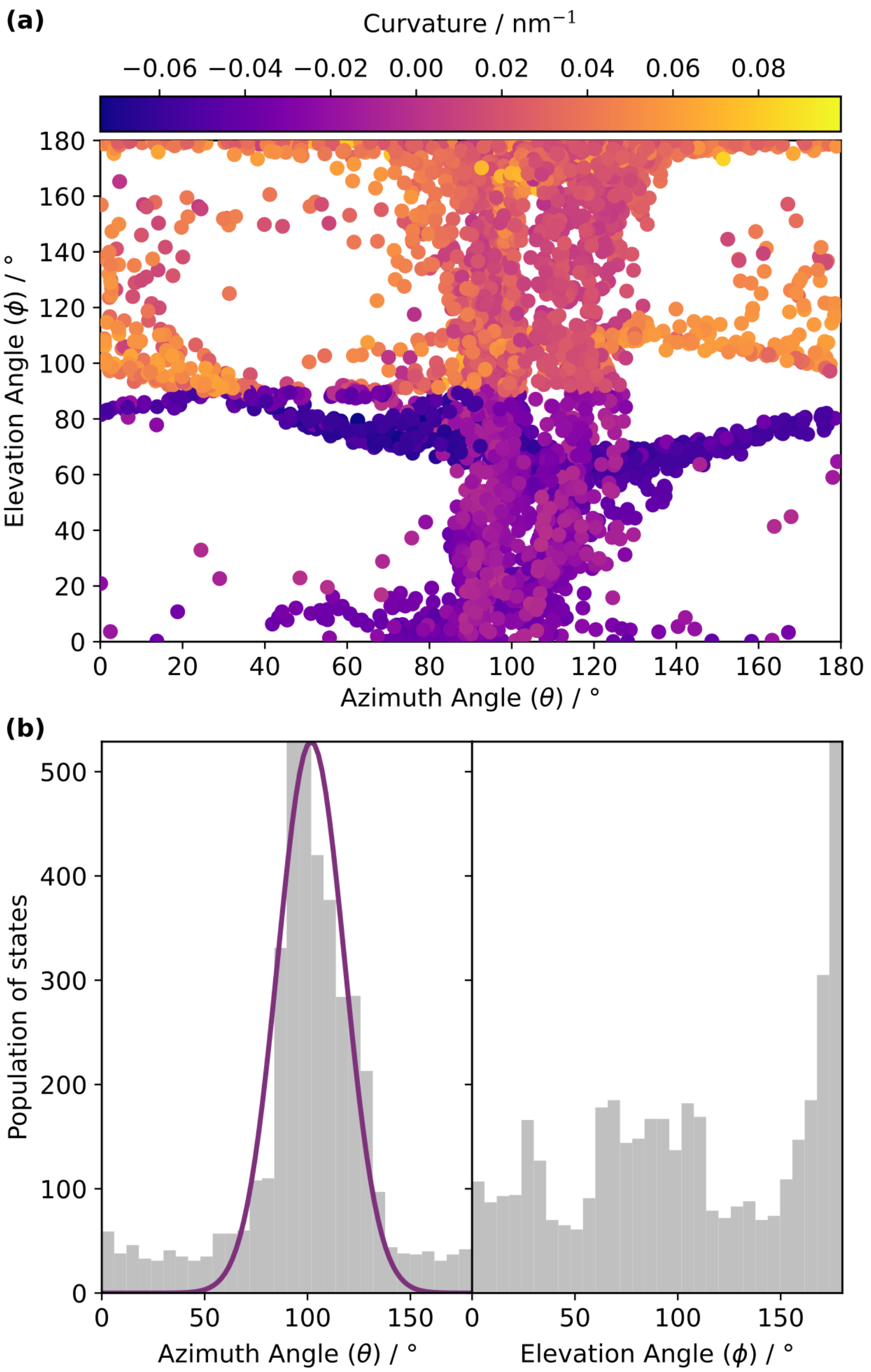

Figure 5.2.7: Predicted orientation of the virion (in spherical coordinates) with respect to the ACE2 membrane, derived from dual binding involving spike proteins in same conformation. Points are coloured according to their associated membrane local curvature. **(a)** Variability in the elevation (ϕ) with azimuth (θ) angle, where each point has been coloured by the appropriate curvature. **(b)** Population of orientational states occupied for each θ and ϕ. A Gaussian has been fitted in palatinate to the θ subfigure.

5.2.3 Binding multiple ACE2 to one spike, or a cross-linked spike, is possible

Recent, unpublished Mass Photometry data obtained by the Kukura group at the University of Oxford, and low-resolution cryo-EM data,[20] suggest that a single spike could bind up to three ACE2 (Figure 5.2.8(a)). In addition, Mass Photometry also indicated the presence of much larger oligomeric complexes, where multiple cross-linked spikes were bound to several ACE2 (Figure 5.2.8(b)). Negative stain-electron microscopy images by Bangaru *et al.* confirmed that the spike on the surface of the virion could cross-link to itself, producing complexes in the MDa.[13] Therefore, we can attempt to build atomic models of both proposed structures to further validate the Mass Photometry data. Of particular interest is whether it is possible to attach two ACE2 to the RBD sites below the cross-linked point (see the bottom pathway in Figure 5.2.8(b)).

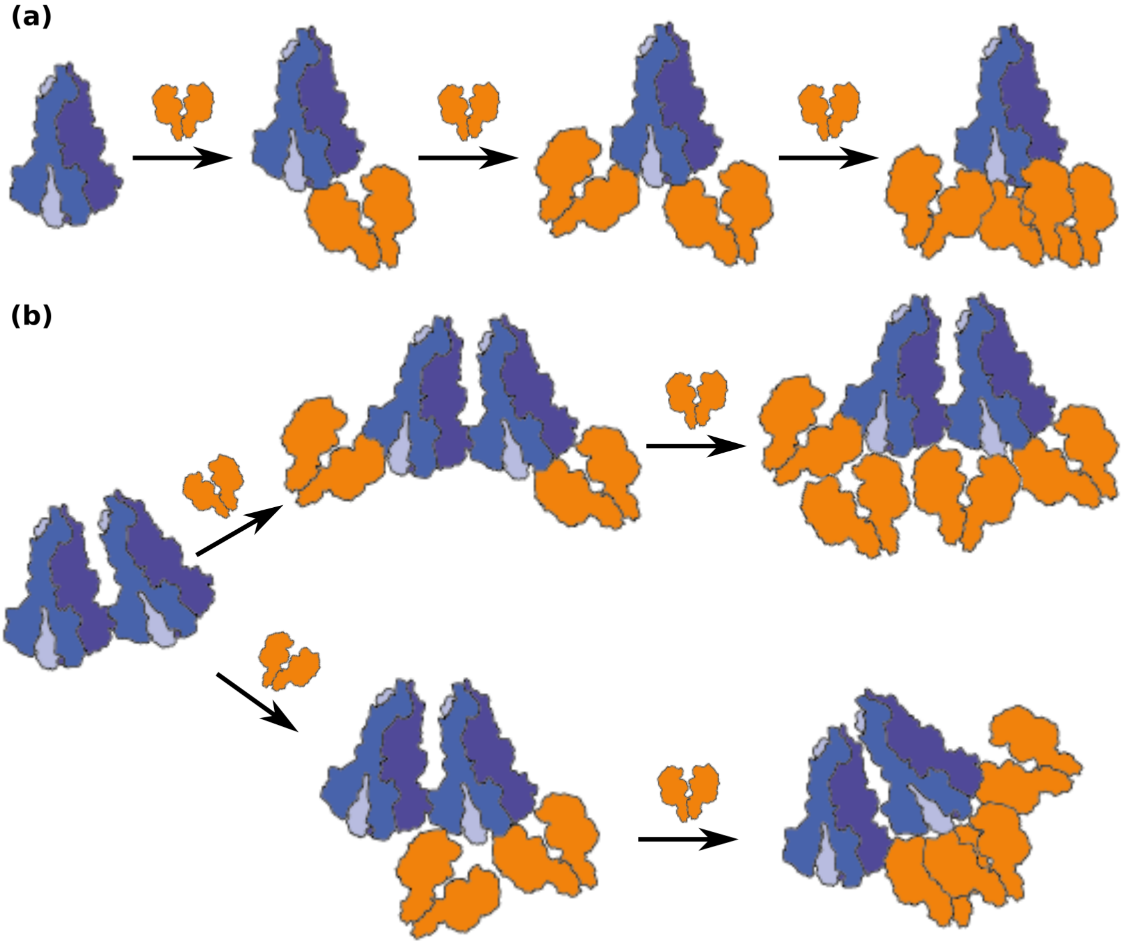

Figure 5.2.8: Schematic of spike-ACE2 oligomeric possible pathways to be considered. **(a)** Formation of three ACE2 bound to a single spike. **(b)** (top pathway) Additional ACE2 avoid the RBDs directly below the cross-linked site, resulting in four ACE2 bound to the spike dimer. (bottom pathway) ACE2 actively bind at every available site, including the two below the cross-linked position, leading to six possible ACE2 bound overall to the spike dimer. Image adapted from our co-authored manuscript with Olerinyova *et al.* (see Publications & Manuscripts).

Aligning the open chain of the SARS-CoV-2 spike glycoprotein 6M18 structure onto the remaining closed chains yielded no conformations that could permit the binding of more than one ACE2. Therefore, we leveraged the MD simulation data of the spike produced by the Amaro research group[14] to generate possible conformations that would facilitate multiple binding of ACE2. Attaching two ACE2 onto the spike, we find that from a total of 1000 frames of a 1 µs simulation, only one featured a conformation that would prevent the backbones from clashing. Applying a Rosetta relaxation refinement[21] followed by a short MD simulation, we were able to generate a structure with two ACE2 bound to the spike with no side-chain clashes. However, adding an additional ACE2 receptor to match the Mass Photometry data at this point was sterically not possible.

Therefore, we utilised the SARS-CoV-1 structure with two chains in the open RBD states (PDB: 6CS1) released by Kirchdoerfer *et al.* in 2018,[22] with a homology of 63%. Following a coordinate transfer of the open chain onto the remaining closed chains, we mutated the structure into the SARS-CoV-2 spike glycoprotein *via* the Modeller program[15] (see Section 5.4.2 for full details). The RBD of SARS-CoV-1 has a large unstructured domain between R306 and I319 (see Figure 5.2.9), in contrast to the equivalent coil in SARS-CoV-2. This allows the spike to adopt an "open flower" configuration where the RBDs are much further apart spatially, as opposed to a "budding flower" configuration in SARS-CoV-2. We could, therefore, align without clashes, one ACE2 per open RBD of this new structure (see Figure 5.2.10).

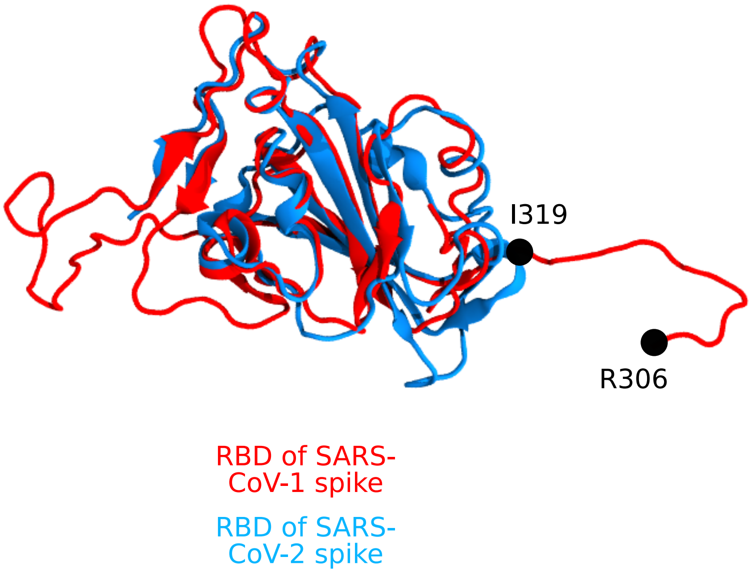

Figure 5.2.9: The unstructured region between R306 and I319 of the RBD keeps the RBDs in an open flower state, thereby facilitating the attachment of the ACE2. The RBD of both SARS-CoV-1 and SARS-CoV-2 are structurally aligned and shown on the right to demonstrate the importance of this unstructured region.

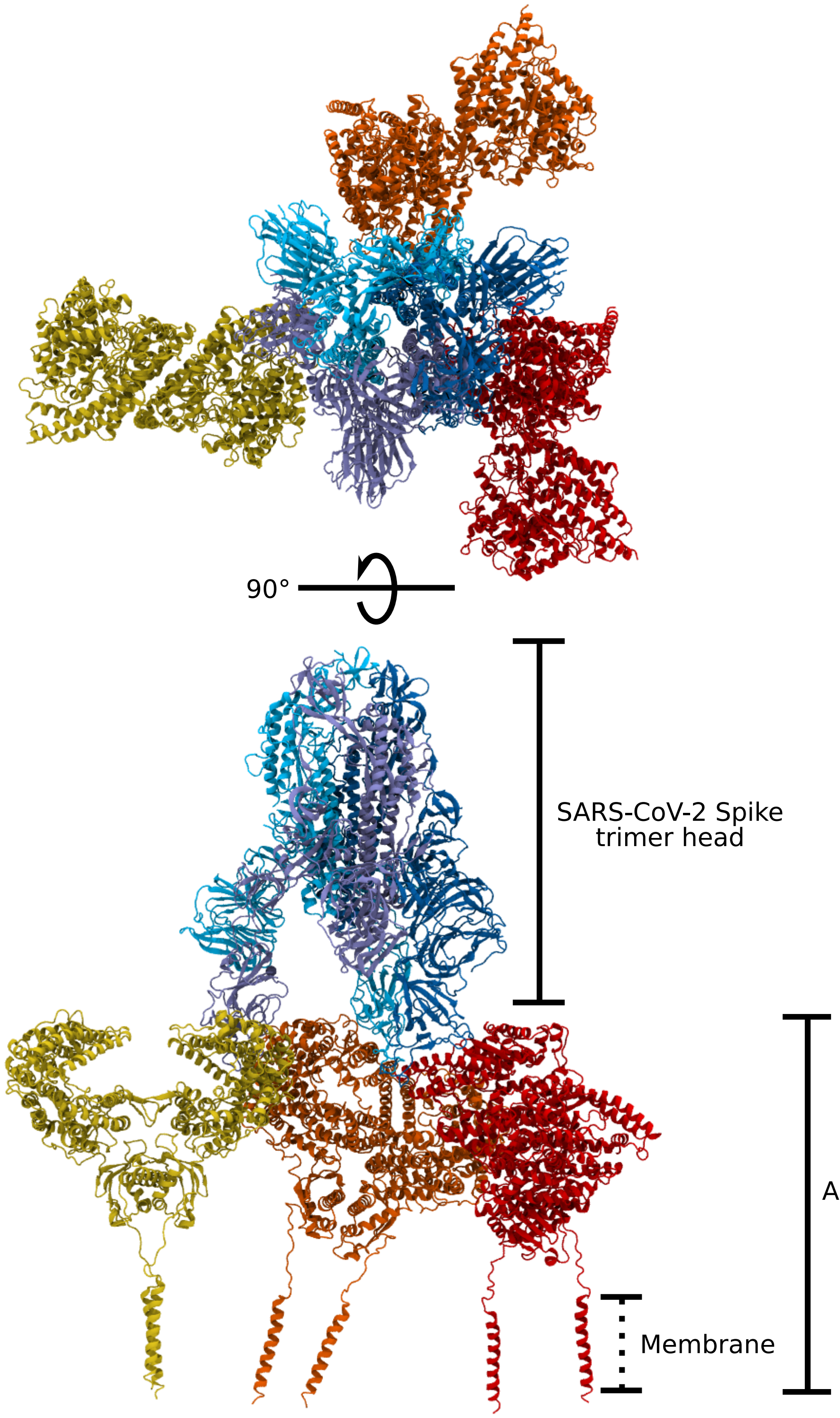

Figure 5.2.10: Three ACE2 bound to a single SARS-CoV-2 spike glycoprotein head. Spike chains are coloured in various shades of blue, while the individual ACE2 are coloured shades of red.

The individual ACE2 receptors in Figure 5.2.10 are at an angle to one another and the membrane due to the symmetry operations required to generate the complex. A large-scale MD simulation would help equilibrate the structure, and allow the ACE2 to settle in the membrane, but this would be very computationally expensive (~4 million atoms), and therefore outside the scope of this work. Defining the central axis of the

ACE2 transmembrane domains as the axis of the membrane (see Section 5.4.1), we return a curvature of 0.003 nm^{-1}, *i.e.* the ACE2 cell membrane is essentially planar.

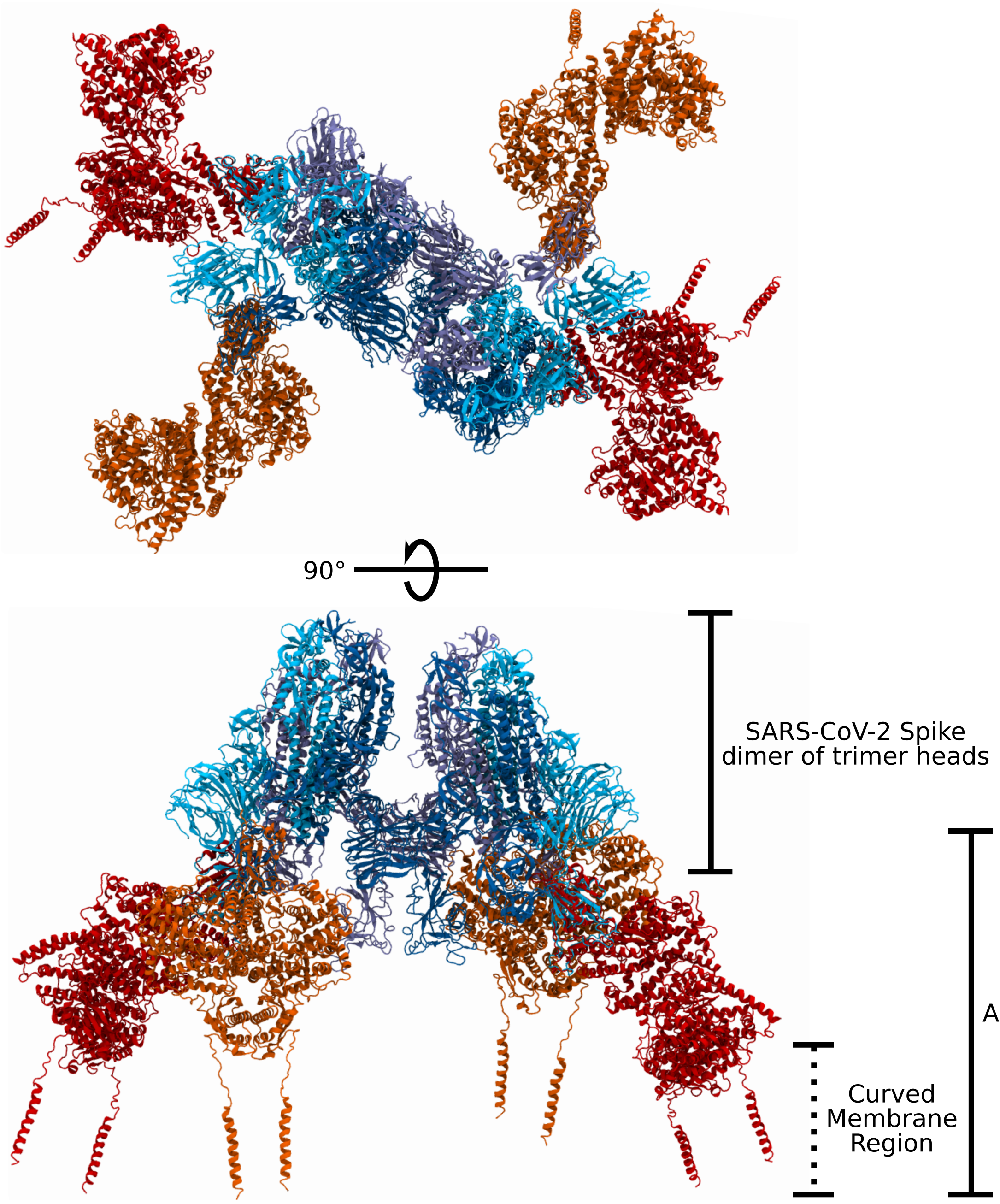

Figure 5.2.11: Four ACE2 bound to the cross-linked SARS-CoV-2 spike glycoprotein head dimer. Spike chains are coloured in various shades of blue, while the individual ACE2 are coloured in red and orange.

Given that two spikes can bind the same ACE2, a single spike can bind three ACE2, and spikes are known to cross-link,[13] there exist several possibilities for the formation of higher-order oligomers that corroborate those detected by the Mass Photometry experiments. We investigated the formation case of a large and specific ACE2-spike complex by leveraging on a recent electron microscopy atomic structure of cross-linked spikes (PDB: 7JJJ).[13] Attaching an ACE2 to every available open RBD resulted in the two central ACE2 (below the cross-linked site) to clash, therefore, the bottom pathway in Figure 5.2.8(b) is not possible. Thus, without invoking any flexibility of RBD domains, up to five ACE2 can bind to the spike dimer. *In vivo*, such an event would require the target cell to feature a local membrane curvature that, although within the range of known curvatures for cells on which ACE2 is expressed,[23] is significant (>0.063 nm^{-1}). By omitting the binding of the central ACE2, such curvature conditions would relax into a gentler one (0.002 nm^{-1}), with a distance between the two sets of ACE2 of 26 nm.

5.2.4 Predicting the ACE2–RBD complex with JabberDock

As a final note, we also tested the feasibility of using JabberDock to dock an integral membrane protein with a soluble protein, by docking the RBD with ACE2. Using the standard transmembrane and soluble pre-processing pipelines separately, we docked the two proteins by keeping the ACE2 fixed and constraining the STID isosurface of the RBD above the plane of the ACE2 cell membrane. From the 300 returned models, only the prediction at rank 196 returned an acceptable result under the CAPRI criteria.[24] However, the rank 4 prediction returned an $f_{\text{nat.}}$ of 0.4, indicating that we had at least partially predicted the binding site.

5.3 Discussion & conclusions

This study first attempted to establish whether the SARS-CoV-2 virion can bind two spike glycoproteins to the ACE2 receptor simultaneously. To this end, we utilised a full-length set of atomistic simulations (equating to 4.2 µs) of the SARS-CoV-2 spike glycoprotein attached to a membrane. These featured a significant range of motion and flexibility due to the stalk attaching the crown of the protein to the transmembrane domain.[14] We aligned these simulations *via* the receptor-binding domain (RBD) to the known bound structure of the RBD with the ACE2 receptor, and then measured the dynamics of the transmembrane domain as a surrogate for the viral membrane. Thus, we should emphasise that this is not a study of the dynamics of the membrane; as the changes in curvature and distances are built from approximations about the local bilayer position given the flexibility of the spike. Instead, we are considering the feasibility of the membrane's shape, and the necessary curvatures it must adopt for a specific conformational binding mode to be possible.

Our models confirm that the dual binding mode onto the ACE2 receptor is possible with no steric clashes, provided the spike is in the open state, which is a necessary prerequisite for binding. By approximating the local shape of the virion's membrane as an arc between the two spike glycoprotein transmembrane regions, we estimate that its curvature can fluctuate between -0.074 nm^{-1} and 0.099 nm^{-1}. 14.3% of simulation time features dual binding models that require a virion's membrane local curvature to be consistent with that found by experiment.[17] In our models, distances between the transmembrane regions of individual spikes range from 16.2 nm to 58.9 nm. Recent tomograms have shown that spikes in the open state are distributed anywhere between 10 nm and 80 nm[18] across the virion surface. Thus, experimental distances measured between spike glycoproteins match those required for dual binding in our models. The SARS-CoV-2 virus has been observed to exhibit a wide range of membrane deformations, creating the higher curvature conditions accommodating an even greater range of possible spike glycoprotein conformations. Neuman *et al.*[25] have noted that a greater concentration of spike glycoproteins are found at regions of high curvature, providing further opportunities for dual binding events.

We find that dual binding events are possible for a multitude of relative arrangements between the virion and the ACE2 host cell. Indeed, many of the predicted conformations require the plane of the virus to be at some angle relative to the normal of the ACE2 cell membrane. We predict that even in the case where the viral membrane becomes locally concave, binding events are still possible, and indeed the curvatures found in these inverted states match those found experimentally.[19] Given this evidence, we can conclude that not only are dual binding events possible but indeed are likely given the density of spike glycoproteins expressed on the virion surface and the number of possible conformations from which they can attach.

With regards to the more extreme angle states discussed in Section 5.2.1 & Section 5.2.2, recent cryo-EM tomography data[18] has shown that the spike head usually adopts a 40° angle with respect to the normal axis of the virion plane. However, within the ~55000 spike glycoproteins manually identified in the tomograms, it was observed that ~150 of them adopted a 90° angle with respect to the normal of the virion membrane. While this may limit the binding modes discussed here, it is worth noting that the more extreme angle states tended towards lower resolution, making for more challenging identification. Therefore, while MD may frequently sample conformations far from the prevailing conformation seen experimentally, we note that these extreme angle states are still physically plausible. Whether it is possible to have two spike glycoproteins in these conformations involved in a dual binding mode, thereby requiring significant local membrane deformation, remains uncertain. Both because of the Cryo-EM indicated stability of these conformations and because of the necessary deformations required of the virion membrane.

We have shown that it is sterically impossible to fit three ACE2 onto the current SARS-CoV-2 spike glycoprotein open structure, and binding two requires a significant number of expensive refinement steps. Yet, Cryo-EM and Mass Photometry data indicate that it must be possible to bind multiple ACE2 to a single spike. In order for a SARS-CoV-2 spike to bind to three ACE2 requires it to adopt a budding flower state similar to available SARS-CoV-1 spike structures.[22] Therefore, we built a homology model of the SARS-CoV-2 spike based on this structure of the SARS-CoV-1 spike[22] to facilitate binding of three ACE2. ACE2 points radially away from the spike, placing the remaining unbound ACE out of reach of any free RBD. This indicates that a single spike should be unable to bind both subunits of the same ACE2.

Given the homology between the SARS-CoV-1 and -2 spikes, it appears plausible that the SARS-CoV-2 spike can exist in the necessary conformation to permit multiple binding, but it might only be possible through an allosteric mechanism triggered by the initial binding of a single ACE2 to a spike. This seems likely given that, in a 1 µs simulation, there were no conformations that featured the RBD in an open flower state. Obtaining a SARS-CoV-2 spike structure in this open flower state might, therefore, pose a considerable challenge. Kirchdoerfer *et al.*'s approach[22] leveraged on a traditional detergent-based approach prior to flash freezing. Whether this is more difficult for the SARS-CoV-2 spike is unknown given the short timeframe of current research, although recent cryo-EM results by Benton *et al.*,[26] coupled with the Mass Photometry data, certainly suggest thats SARS-CoV-2 spike must be capable of adopting a similar conformation. Our computational approach, therefore, has assisted in alleviating this knowledge gap. Furthermore, we have used the dynamics of the

SARS-CoV-2 spike, or rather the absence of dynamics, to highlight the necessity of some underlying mechanism to facilitate multiple binding of ACE2 to a single spike.

We then assessed the possible oligomeric complexes when the spike cross-links to itself. Leveraging once again on the homolog based on the SARS-CoV-1 spike and a recent structure of the spike dimer (PDB: 7JJJ),[13] we find that up to five ACE2 are able to bind to free RBDs without steric clashes. To reduce any curvature-based energetic penalties on the ACE2 cell, we can remove the central ACE2 below the cross-linked site, leaving 4 ACE2 bound. This agrees with preliminary Mass Photometry data suggests that up to four ACE2 can bind to the cross-linked dimer at a time. Adding an additional spike to create a trimer of trimers, would close off either one or two of the RBD sites directly below the new cross-linked site depending on the curvature of the ACE2 membrane. In other words, for every spike added to the oligomer, an additional one or two ACE2 dimers are able to bind. Therefore, we can expect the number of ACE2 receptors to grow as n+2 for an n-length oligomer, or 2n+1 if we allow higher curvatures. Realistically, forming a large oligomeric complex *in vivo* might be challenging, as it requires a high density of ACE2, with the found high oligomeric ACE2 states occuring only at high concentrations of ACE2 in the Mass Photometry experiments. Furthermore, adding one or two ACE2 to a cross-linked spike decreases the number of entry points or pathways for an additional ACE2, particularly as they are constrained to the surface of the cell. Perhaps more importantly, given the helical twist exhibited by the spike oligomer as it grows, the significant torsion angles between terminal spikes would enforce a continual strain on the ACE2 cell membrane. Thus, while these models may be sterically plausible, further computational modelling is required to confirm the likelihood of their formation. Nevertheless, our results have demonstrated the plethora of possible oligomeric ACE2-spike complexes that can form. Indeed, recent work by Barros *et al.*, have highlighted the inherent flexibility of the ACE2 receptor[27] in complex with the RBD, mirroring the range of motion exhibited by the spike protein. We expect this flexibility to facilitate the variety of oligomeric states discussed here while decreasing any high curvature strains for both the virus and cell membranes.

The inability of JabberDock to provide a high ranking, successful prediction, is likely, in part, to the absence of the rest of the SARS-CoV-2 spike head. Its presence would render many of the arrangements sterically impossible. Better insight could be achieved by using the spike head, with the ACE2 restricted to move in the xy-plane, using the transmembrane protein docking routine outlined in Section 3.4. However, this would be very computationally costly. Further tests of JabberDock's feasibility with similar, but smaller, docking problems are required before a conclusion can be made on its suitability.

In summary, our key findings are:

1. Two spikes can bind to a single ACE2 at a time with a range of virion curvatures.
2. The virion can approach the prey cell from a host of different orientations, and still enable successful binding of two spikes to the single ACE2.
3. Current, microsecond timescale simulations of the open SARS-CoV-2 spike provide no conformations that enable three ACE2 to bind to a spike simultaneously. However, a homolog based on the spike of SARS-CoV-1 does provide configurations that enable three ACE2s to attach. Thus, to agree with the experimental data there must be some domain reshuffling that extends the RBD into the open flower state following a single binding event, facilitating further spike-ACE2 interactions.
4. In the event that n spikes cross-link, it is not possible to bind two ACE2 at the two RBDs below the cross-linked site. While one may bind below these cross-linked regions, there is the caveat that this imposes a significant strain on the ACE2 cell membrane. Therefore, we expect the number of ACE2 to grow as n+2 with n SARS-CoV-2 spike glycoproteins, which agrees with preliminary Mass Photometry data.

Thus, from the dynamics extracted from a large MD simulation, we have been able to draw several conclusions that are essential to the wider understanding into how SARS-CoV-2 is able to penetrate into a cell. The number of binding modes available to the SARS-CoV-2 spike glycoprotein is extensive, which goes some way to explaining its potency, highlighting why it has caused one of the world's worst pandemics of the last one hundred years. Yet, it also opens the door to possible medical targeting. The propensity of the spikes to form into large cross-linked species could be used as a route towards therapeutics that harness this behaviour as a means to aggregate and remove the spikes, preventing entry into the cell.

5.4 Methods

5.4.1 Modelling of ACE2 bound to two spikes

The binding of two spike glycoproteins onto the ACE2 receptor is dependent on three primary aspects: the conformation of the spike glycoprotein, the position of the virion over the ACE2 receptor and the local shape of the viral membrane (see graphical representation in Figure 5.2.2). To generate a template representing two spike proteins bound to the same ACE2, we leveraged on a crystal structure of the SARS-CoV-2 spike RBD bound to the soluble domain of monomeric ACE2 (PDB: 6M0J). We duplicated and aligned 6M0J *via* its ACE2 atoms to each of the chains in the full ACE2 homodimer (PDB: 6M18), obtaining a complete ACE2 receptor with two SARS-CoV-2 spike RBD domains bound. To represent the flexibility of SARS-CoV-2 spike in our models, we adopted the 4.2 µs-long simulation made available by the Amaro group, featuring a fully glycosylated spike protein with one open RBD, embedded in a lipid bilayer.[14] We duplicated each of its 4200 simulated spike conformations (one per ns) and aligned them *via* their open RBD (chain A) to our template RBD domains bound to the complete ACE2 homodimer receptor (see examples of resulting models in Figure 5.2.4).

From the relative arrangement of the ACE2 receptor and its two bound spikes, we inferred, for each spike conformation, the relative position of the virion and the curvature of the membrane between the transmembrane regions of its two spikes. To this end, we defined our reference frame as the ACE2 cell membrane, where the z-axis is the membrane plane normal, and the origin is at the mean position of the ACE2 receptor membrane-entry and -exit atoms (Cα atoms of residues 741 to 761). We then defined the orientation of the viral membrane at the two spikes transmembrane regions according to the axes connecting the mean positions of their membrane-entry and -exit residues (Cα atoms residues 1214 and 1234 for all three chains). By correcting these axes by the observed transmembrane region tilt, we obtained normal vectors defining planes parallel to the simulated viral membrane. Finally, we defined the viral membrane shape according to the arc connecting the two viral membrane planes, assuming that the membrane has constant curvature between the two transmembrane regions. From all these definitions, we derived the angle of attack of the virion according to the vector connecting the origin to the midpoint of the arc between the spike proteins, represented in spherical coordinates (elevation ϕ and azimuth θ). The distance between spikes' transmembrane regions was measured as the length of the arc connecting them. The following provides the full details on our methodology to determine ϕ, θ and the local membrane curvature:

1. To account for the spike's transmembrane tilt, we first record, at every frame,

the rotational matrix required to project the principal axis of the transmembrane domain onto the normal to the plane of the viral bilayer. These matrices are non-commutative with the rotational alignment matrix used to generate the spike-ACE2 complex; hence we convert these corrective matrices into intrinsic rotations, which are invariant under a change of coordinate system.

2. Following alignment of the spikes to the RBD bound to the ACE2 receptor, we define the central axis for each spike transmembrane domain, corrected for tilt *via* the intrinsic rotation matrix, as the normal to the local viral membrane. (see Figure 5.2.4). By projecting the transmembrane region's axis onto the known viral membrane normal, we create the condition where the local viral membranes exactly bisect these principal axes at the midpoint (*i.e.* our planes, by definition, are parallel to the simulated viral membrane) – see Figure 5.2.4.

3. Three points: two from the individual average positions of the spike transmembrane domains (central blue pseudo-atom in Figure 5.2.4), and one from the intersection between the two planes, defines an arc from which the curvature of the viral membrane can be derived. The curvature is considered nominal (resulting in a convex virion) when the normal to the tangent at the top of the arc, $\vec{n}_\mathrm{v}$, is aligned along the inner to outer leaflet of the viral membrane. It is considered inverse (concave virion) when the opposite is true.

4. The arc length connecting the transmembrane domains *via* the intersection defines the maximum possible geodesic distance between the transmembrane domains. The point-to-point distance between them represents the case where curvature tends to zero, representing the minimum distance.

5. $\vec{n}_\mathrm{v}$ is used to examine the virion's orientation relative to the reference frame of the ACE2 cell, defining the azimuth (θ) and elevation (ϕ) angles as $\vec{n}_\mathrm{v}$ *versus* the positive y- and z-axes respectively.

6. The normal to the ACE2 cell membrane ($\vec{n}_\mathrm{c}$) is colinear to the positive z-axis, making the two equivalent. When the z component of $\vec{n}_\mathrm{v}$ is aligned along the negative z-axis we obtain the antiparallel ϕ state (Figure 5.2.1(1,2), right conformation in Figure 5.2.4), when it is parallel to the positive z-axis we obtain the parallel ϕ state (Figure 5.2.1(3,4), left conformation in Figure 5.2.4). $\phi = 90°$ indicates that the viral membrane is perpendicular to the ACE2 membrane. We can, in principle, have either convex or concave membranes in both of these states.

5.4.2 Modelling of multiple ACE2 bound to cross-linked spikes

Following a mapping of the open RBD chain of the SARS-CoV-2 spike glycoprotein onto the remaining closed chains, we found it was sterically not possible to attach additional ACE2. We, therefore, leveraged on the SARS-CoV-1 structure with two chains in the open RBD states (PDB: 6CS1), with a homology of 63%.[22] Following an alignment of the open state chain A onto the remaining closed chains, we mutated the structure into the SARS-CoV-2 spike glycoprotein *via* the Modeller program.[15] Missing motifs up to 10 residues in length were patched, while the structure was kept frozen to prevent optimisation away from the desired conformational state. We could align, without clashes, one ACE2 per open RBD in this new structure (see Figure 5.2.10).

To examine the plausibility of oligomeric states involving cross-linked spikes (see Figure 5.2.11), we aligned our homology model onto an available dimer of the SARS-CoV-2 spike (PDB: 7JJJ). We were then able to align five ACE2 onto the free binding sites without clashes. To calculate the required cellular membrane curvature, we first measured the curvature between two ACE2 bound to the same spike protein using the same protocol established above. As curvature between the two was only 0.01 nm^{-1}, we approximated the two transmembrane stalks as parallel and took their average principal axes centred at their midpoint as the starting point for calculating the curvature of the cellular membrane. In calculating the curvature of the two spike – five ACE2 oligomer, instead of calculating the intersection of two planes, we define the principal axis of the central ACE2 transmembrane stalk as the normal to the intersection point. The calculation was performed on a single frame, rather than a simulation, hence the absence of a standard deviation in the reported value.

5.5 Bibliography

[1] Y. Guan, B. J. Zheng, Y. Q. He, X. L. Liu, Z. X. Zhuang, C. L. Cheung, S. W. Luo, P. H. Li, L. J. Zhang, Y. J. Guan, K. M. Butt, K. L. Wong, K. W. Chan, W. Lim, K. F. Shortridge, K. Y. Yuen, J. S. Peiris and L. L. Poon, *Isolation and characterization of viruses related to the SARS coronavirus from animals in Southern China*, *Science (80-.).*, 2003, **302**, 276–278.

[2] W. Li, Z. Shi, M. Yu, W. Ren, C. Smith, J. H. Epstein, H. Wang, G. Crameri, Z. Hu, H. Zhang, J. Zhang, J. McEachern, H. Field, P. Daszak, B. T. Eaton, S. Zhang and L. F. Wang, *Bats are natural reservoirs of SARS-like coronaviruses*, *Science (80-.).*, 2005, **310**, 676–679.

[3] C. Kamps, B. S. & Hoffmann, *SARS Reference — Preface*, 2003, `http://www.sarsreference.com/sarsref/preface.htm`.

[4] A. M. Zaki, S. van Boheemen, T. M. Bestebroer, A. D. Osterhaus and R. A. Fouchier, *Isolation of a Novel Coronavirus from a Man with Pneumonia in Saudi Arabia*, *N. Engl. J. Med.*, 2012, **367**, 1814–1820.

[5] H. M. Al-Dorzi, M. D. Van Kerkhove, J. S. Peiris and Y. M. Arabi, *Middle east respiratory syndrome coronavirus*, *ERS Monogr.*, 2016, **2016**, 21–34.

[6] C. Huang, Y. Wang, X. Li, L. Ren, J. Zhao, Y. Hu, L. Zhang, G. Fan, J. Xu, X. Gu, Z. Cheng, T. Yu, J. Xia, Y. Wei, W. Wu, X. Xie, W. Yin, H. Li, M. Liu, Y. Xiao, H. Gao, L. Guo, J. Xie, G. Wang, R. Jiang, Z. Gao, Q. Jin, J. Wang and B. Cao, *Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China*, *Lancet*, 2020, **395**, 497–506.

[7] L. L. Ren, Y. M. Wang, Z. Q. Wu, Z. C. Xiang, L. Guo, T. Xu, Y. Z. Jiang, Y. Xiong, Y. J. Li, X. W. Li, H. Li, G. H. Fan, X. Y. Gu, Y. Xiao, H. Gao, J. Y. Xu, F. Yang, X. M. Wang, C. Wu, L. Chen, Y. W. Liu, B. Liu, J. Yang, X. R. Wang, J. Dong, L. Li, C. L. Huang, J. P. Zhao, Y. Hu, Z. S. Cheng, L. L. Liu, Z. H. Qian, C. Qin, Q. Jin, B. Cao and J. W. Wang, *Identification of a novel coronavirus causing severe pneumonia in human: a descriptive study*, *Chin. Med. J. (Engl).*, 2020, **133**, 1015–1024.

[8] N. Zhu, D. Zhang, W. Wang, X. Li, B. Yang, J. Song, X. Zhao, B. Huang, W. Shi, R. Lu, P. Niu, F. Zhan, X. Ma, D. Wang, W. Xu, G. Wu, G. F. Gao and W. Tan, *A Novel Coronavirus from Patients with Pneumonia in China, 2019*, *N. Engl. J. Med.*, 2020, **382**, 727–733.

[9] P. Zhou, X. L. Yang, X. G. Wang, B. Hu, L. Zhang, W. Zhang, H. R. Si, Y. Zhu, B. Li, C. L. Huang, H. D. Chen, J. Chen, Y. Luo, H. Guo, R. D. Jiang, M. Q. Liu, Y. Chen, X. R. Shen, X. Wang, X. S. Zheng, K. Zhao, Q. J. Chen, F. Deng, L. L. Liu, B. Yan, F. X. Zhan, Y. Y. Wang, G. F. Xiao and Z. L. Shi, *A pneumonia outbreak associated with a new coronavirus of probable bat origin*, *Nature*, 2020, **579**, 270–273.

[10] *WHO Coronavirus Disease (COVID-19) Dashboard — WHO Coronavirus Disease (COVID-19) Dashboard*, https://covid19.who.int/.

[11] D. Wrapp, N. Wang, K. S. Corbett, J. A. Goldsmith, C. L. Hsieh, O. Abiona, B. S. Graham and J. S. McLellan, *Cryo-EM structure of the 2019-nCoV spike in the prefusion conformation*, *Science (80-.).*, 2020, **367**, 1260–1263.

[12] R. Yan, Y. Zhang, Y. Li, L. Xia, Y. Guo and Q. Zhou, *Structural basis for the recognition of SARS-CoV-2 by full-length human ACE2*, *Science (80-.).*, 2020, **367**, 1444–1448.

[13] S. Bangaru, G. Ozorowski, H. L. Turner, A. Antanasijevic, D. Huang, X. Wang, J. L. Torres, J. K. Diedrich, J.-H. Tian, A. D. Portnoff, N. Patel, M. J. Massare, J. R. Yates, D. Nemazee, J. C. Paulson, G. Glenn, G. Smith and A. B. Ward, *Structural analysis of full-length SARS-CoV-2 spike protein from an advanced vaccine candidate*, *Science (80-.).*, 2020, **370**, 1089–1094.

[14] L. Casalino, Z. Gaieb, J. A. Goldsmith, C. K. Hjorth, A. C. Dommer, A. M. Harbison, C. A. Fogarty, E. P. Barros, B. C. Taylor, J. S. Mclellan, E. Fadda and R. E. Amaro, *Beyond shielding: The roles of glycans in the SARS-CoV-2 spike protein*, *ACS Cent. Sci.*, 2020, **6**, 1722–1734.

[15] A. Fiser and A. Sali, *ModLoop: Automated modeling of loops in protein structures*, *Bioinformatics*, 2003, **19**, 2500–2501.

[16] A. Roy, A. Kucukural and Y. Zhang, *I-TASSER: A unified platform for automated protein structure and function prediction*, *Nat. Protoc.*, 2010, **5**, 725–738.

[17] N. Chen, M. Zhou, X. Dong, J. Qu, F. Gong, Y. Han, Y. Qiu, J. Wang, Y. Liu, Y. Wei, J. Xia, T. Yu, X. Zhang and L. Zhang, *Epidemiological and clinical characteristics of 99 cases of 2019 novel coronavirus pneumonia in Wuhan, China: a descriptive study*, *Lancet*, 2020, **395**, 507–513.

[18] H. Yao, Y. Song, Y. Chen, N. Wu, J. Xu, C. Sun, J. Zhang, T. Weng, Z. Zhang, Z. Wu, L. Cheng, D. Shi, X. Lu, J. Lei, M. Crispin, Y. Shi, L. Li and S. Li, *Molecular architecture of the SARS-CoV-2 virus*, *Cell*, 2020, **183**, 730–738.

[19] C. Liu, Y. Yang, Y. Gao, C. Shen and B. Ju, *Viral Architecture of SARS-CoV-2 with Post-Fusion Spike Revealed by Cryo-EM*, *bioRxiv*, 2020, 1–17.

[20] T. M. Clausen, D. R. Sandoval, C. B. Spliid, J. Pihl, H. R. Perrett, C. D. Painter, A. Narayanan, S. A. Majowicz, E. M. Kwong, R. N. McVicar, B. E. Thacker, C. A. Glass, Z. Yang, J. L. Torres, G. J. Golden, P. L. Bartels, R. N. Porell, A. F. Garretson, L. Laubach, J. Feldman, X. Yin, Y. Pu, B. M. Hauser, T. M. Caradonna, B. P. Kellman, C. Martino, P. L. Gordts, S. K. Chanda, A. G. Schmidt, K. Godula, S. L. Leibel, J. Jose, K. D. Corbett, A. B. Ward, A. F. Carlin and J. D. Esko, *SARS-CoV-2 Infection Depends on Cellular Heparan Sulfate and ACE2*, *Cell*, 2020, **183**, 1043–1057.e15.

[21] L. G. Nivón, R. Moretti and D. Baker, *A Pareto-Optimal Refinement Method for Protein Design Scaffolds*, *PLoS One*, 2013, **8**, e59004.

[22] R. N. Kirchdoerfer, N. Wang, J. Pallesen, D. Wrapp, H. L. Turner, C. A. Cottrell, K. S. Corbett, B. S. Graham, J. S. McLellan and A. B. Ward, *Stabilized coronavirus spikes are resistant to conformational changes induced by receptor recognition or proteolysis*, *Sci. Rep.*, 2018, **8**, 1–11.

[23] J. Zimmerberg and M. M. Kozlov, *How proteins produce cellular membrane curvature*, *Nat. Rev. Mol. Cell Biol.*, 2006, **7**, 9–19.

[24] M. F. Lensink, R. Méndez and S. J. Wodak, *Docking and scoring protein complexes: CAPRI 3rd Edition*, *Proteins Struct. Funct. Bioinforma.*, 2007, **69**, 704–718.

[25] B. W. Neuman, G. Kiss, A. H. Kunding, D. Bhella, M. F. Baksh, S. Connelly, B. Droese, J. P. Klaus, S. Makino, S. G. Sawicki, S. G. Siddell, D. G. Stamou, I. A. Wilson, P. Kuhn and M. J. Buchmeier, *A structural analysis of M protein in coronavirus assembly and morphology*, *J. Struct. Biol.*, 2011, **174**, 11–22.

[26] D. J. Benton, A. G. Wrobel, P. Xu, C. Roustan, S. R. Martin, P. B. Rosenthal, J. J. Skehel and S. J. Gamblin, *Receptor binding and priming of the spike protein of SARS-CoV-2 for membrane fusion*, *Nature*, 2020, **588**, 327.

[27] E. P. Barros, L. Casalino, Z. Gaieb, A. C. Dommer, Y. Wang, L. Fallon, L. Raguette, K. Belfon, C. Simmerling and R. E. Amaro, *The Flexibility of ACE2 in the Context of SARS-CoV-2 Infection*, *Biophys. J.*, **accepted**, year.

6 | Conclusions

In this thesis, we have studied the importance of accomodating protein dynamics in protein quaternary structure prediction as a means to understand the function of a protein. In Chapter 1, we outlined three key aims to be examined as part of this study. Each of these aims has been addressed through the course of this thesis. Herein, we discuss their fulfilment and how we can build upon the work that has been so far accomplished.

6.1 Summary

6.1.1 Aim 1: To develop a representation that encompasses the key characteristics of a protein

The STID map was originally designed with the forethought of applying it exclusively to protein-protein docking. Ideally, however, any model used in such a context should, independently, demonstrate that it represents the physical characteristics of a protein. Once the benchmarks in Section 2.3 revealed that this was indeed the case, in contrast to other coarse-grained models,[1] it became obvious that we should also establish the STID map's utility in a host of other scenarios. Chapter 2 addresses these applications.

In Section 2.4 we used the STID map construction procedure, truncated at Equation 2.2.3, to predict a series of dielectric permittivities for a selection of solvents. A computational method capable of predicting the variation in permittivity on a nanometre scale would prove invaluable in the study of interfaces, both between solvents and protein/solvent. While our method does not outperform existing methods to predict dielectric variation on the nanoscale,[2] **its ability to correctly identify trends is encouraging, as it indicates that the underpinning theory works** in practice.

In Section 2.5 we addressed the controversy surrounding the true density of a protein, as described in Chapter 1. Given that the consensus value for a protein (1.35 g cm^{-3}), is largely based on volumes generated from crystal structures,[3] yet is applied in a variety of *in vivo* and *in vitro* based studies, we endeavoured to provide better clarity to its true value and its dependence on the surrounding environment. **We identified a relationship between protein dynamics and density**, matching recent experimental evidence by Ashkarran *et al.* in solution,[4] and *in vacuo* ion mobility data.[5] These results demonstrate that the STID map isosurface, derived through the independent benchmark in Section 2.3.4, is **capable of representing the key properties of a protein in different environments through accommodation of side-chain dynamics**.

Finally, in Section 2.6 we demonstrated that a STID map isosurface's topography is sensitive to the binding partner. Indeed, we showed in Section 4.3 that mutations could also significantly influence the topography. This characteristic allowed us to track which amino acids are affected by some distal binding interaction. These results reveal that **STID maps are sensitive to mutagenesis and binding interactions**, with the exhibited dynamics of individual residuals, fundamentally altered.

Therefore, we have shown that STID maps are well-grounded in physical theory and represent some critical properties of a protein, such as its density, and are sensitive to small changes in the structure or sequence. Thus, we are confident that **STID maps are a sufficient model of a protein's structure and local dynamics.**

6.1.2 Aim 2: To apply our representation in a protein-protein docking context

Having established a physical grounding for our model, we then applied the STID map isosurface in a protein-protein docking context. Specifically, we designed a protein docking engine called JabberDock (`https://github.com/degiacom/JabberDock`), which we then applied to both a water-soluble and transmembrane protein docking benchmark, before demonstrating its utility in two applications. This was the focus of the work in Chapter 3. Following this, we wished to investigate further means to enhance the algorithm by using bound surrogates for unbound docking and improving the method by which the STID maps are applied and the optimisation approach. This was the motivation for Chapter 4.

We first applied JabberDock in Section 3.3 to the standard CAPRI benchmark of 224 water-soluble complexes.[6] The cases are split by difficulty depending on backbone flexibility. JabberDock returned an **acceptable success rate of 56.3%, 60.0% and 54.9% for the easy, medium and hard cases** respectively, and a corresponding intermediate rate of 29.2%, 22.2%, and 25.8%. Boundary conditions were associated with either a flat binding site, that the scoring function failed to score highly, or occluded interfaces, which the optimiser struggled to navigate into. **This work's key**

success is the absence of any deterioration in the success rate with flexibility, indicating that accomodating side-chain dynamics into the model is critical to a successful docking engine. JabberDock is, therefore, very competitive *versus* other methods.

In Section 3.4 we examined JabberDock's effectiveness in transmembrane protein docking by predicting a series of 20 unbound α-helical complexes sourced from three different benchmarks.[7–9] For this dataset, we returned **acceptable and intermediate success rates of 75% and 40%**, respectively. Thus, our inclusion of differential side-chain dynamics that naturally arise from different environments *via* MD, coupled with the restricted conformational space imposed by the membrane, facilitates enhanced quality docking. These results place **JabberDock as highly competitive in the transmembrane protein docking field, without the need for any manual preprocessing.**

With these benchmarks in toe, in the latter half of Chapter 3, we used JabberDock in two novel applications. In Section 3.5.1 we used JabberDock to predict TTR fibril formation from a starting monomeric unit of the TTR tetramer, generating atomistic fibril structures that matched electron micrographs. Section 3.5.2 illustrated that we could predict the loss of detergent given the formation of a bo_3 oxidase dimer, a complex whose presence was not yet established in the literature. Our results corroborated Mass Photometry experimental data and confirmed the application of both Mass Photometry and JabberDock in transmembrane protein structure prediction.

In Section 4.3 we attempted to qualify whether the use of bound surrogates could enhance the overall quality of an unbound target; that is, that extracting the dynamics of a ligand or receptor bound to a foreign binding partner could improve its docking with its intended target partner. Our results indicated that **a preliminary step of searching for surrogates, and their application thereof, should yield higher quality predictions**, and perhaps assist in addressing some of JabberDock's limitations.

Section 4.4 outlined our current efforts to improve the current implementation of the STID map isosurfaces by converting them into a Signed Distance Function, an object used frequently in Computer Graphics design. The Signed Distance Function provides the distance to the isosurface at every point in space, thereby trivialising access to gradients in a gradient-descent based minimisation procedure. It also predicates the derivation of an analytically derived scoring function. While **early results indicated that the two aforementioned boundary conditions of JabberDock could be addressed with this approach**; due to the necessary intersection between STID isosurfaces at the known bound position, the results did not improve upon the current version of JabberDock. Furthermore, the applied clustering algorithms demonstrated that convergence was difficult to achieved due to the number of solutions returned. These concerns highlighted the need for a rapid optimisation method.

Our first step towards this is the focus of Section 4.5 which described an outer-loop rapid optimisation method based on ray-tracing. This approach marches a ligand to the receptor by the bounds of its distance to it, thereby efficiently, and swiftly, generating docked solutions. **Compared with our previous efforts, this Shape Tracing optimisation method is ~400× faster on the same hardware.** Therefore, we need to couple this technique with a second, inner-loop gradient-based optimisation procedure to achieve an improved version of the JabberDock algorithm.

Therefore, we have shown that an isosurface developed from a STID map has remarkable utility in a protein-protein docking context, both for water-soluble and for the more challenging transmembrane proteins. The JabberDock tool is continually maintained and is freely accessible for users, with an accompanying tutorial and manual. We foresee the exploitation of JabberDock in a wide range of novel applications, as we have done, thereby enhancing research in the quaternary structure prediction arena, particularly for those looking to predict integral membrane protein complexes.

6.1.3 Aim 3: To leverage Molecular Dynamics simulations to elucidate diverse quaternary structure

Aside from accounting for side-chain dynamics, we also endeavoured to showcase how harnessing larger conformational dynamics extracted from MD can facilitate protein structure prediction. The recent emergence, and subsequent need for research into the SARS-CoV-2 virology, provided an opportunity for this. In the wake of the COVID-19 pandemic, we, therefore, endeavoured to uncover the diverse, multiple oligomeric states of SARS-CoV-2 spike glycoprotein and the ACE2 receptor, in collaboration with an experimental group at Oxford. The goal of this work was to understand the various modes of attack for the virus in the pursuit of therapeutics.

Utilising a 4.2 µs simulation of the full-length SARS-CoV-2 spike glycoprotein produced by the Amaro lab,[10] we were able to determine that two spike proteins could attach to a single ACE2 receptor from numerous conformational states. With these states in mind, we could predict the necessary curvatures needed of the local virion membrane to accommodate the binding action. A key result from this is that **the spike proteins spend a significant amount of time in conformations that correspond to the nominal spherical curvature typically exhibited by the virion.** In addition, only minor deformations are required of the membrane outside of this time. The MD data also indicate the various positions the virus particle can take with respect to the ACE2 cell during its attack.

We then assessed the various oligomeric states possible between multiple ACE2 and the spike glycoprotein, for both a single and cross-linked spike trimer. We were unable to align multiple ACE2 onto the Amaro lab's[10] simulated spike glycoprotein, highlighting that some conformational change must occur following binding of the first ACE2. *Via* a homology model derived from the SARS-CoV-1 spike glycoprotein, **we were able to align three ACE2 into the single spike's known binding positions.** For the spike dimer, we found that up to five ACE2 could attach, with the stipulation that this may place a significant strain on the ACE2 cell membrane. With this and the Mass Photometry data in mind, **we expect up to four ACE2 could attach to the spike dimer, with the number of ACE2 increasing with $n+2$ for an n-mer spike oligomer.**

Therefore, we have been able to predict a plethora of oligomeric states that can form between the spike glycoprotein and ACE2, both using a large MD simulation (accomplishing Aim 3), but also using homolog structures were necessary in the absence of simulation data. An extensive simulation of the full-length spike glycoprotein bound to the ACE2 is undoubtedly of interest in this context, but it is prohibitively expensive.

6.1.4 The impact of dynamics in protein assembly

We have realised the three aims laid out in Section 1.5, corresponding to a study into the impact of dynamics in protein quaternary structure prediction. Sections 2.3.4, 2.4, and 2.5 were used to establish the physical ground of our STID map; demonstrating that it was physically well-grounded, and a good representation for protein shape, electrostatics and, most importantly for our goals, local dynamics. We then applied our maps in a protein-protein docking context through our docking engine, JabberDock, in Sections 3.3 and 3.4, showing it could be used for both water- and lipid-soluble proteins. This lead to the direct implementation of JabberDock to two applications in Sections 3.5.1 and 3.5.2. Overcoming the identified boundary conditions found through our docking benchmarks was the focus of Sections 4.4 and 4.5. Furthermore, by demonstrating the sensitivity of the STID map to a protein's microenvironment in Section 2.6, we opened another possible avenue to address these boundary conditions through harnessing surrogate bound dynamics in Section 4.3. Indeed, Section 2.6 suggested that isosurfaces could reflect the impact of mutagenesis on protein-protein interactions; Section 4.3 subsequently confirmed this by demonstrating that JabberDock, correctly, was unable to predict the bound complex given an unstable, mutated ligand. Despite JabberDock's success, particularly as it pertains to more flexible cases, there is a lack of support for accomodating larger, conformational dynamics; in both the docking engine and the STID maps. Therefore, we endeavoured to exploit large-scale MD simulations in Chapter 5 to enhance protein quaternary structure prediction. Merging the work of the aforementioned Sections and the work in Chapter 5 is a major challenge, but one that, if successful, would be of great benefit to the biomolecular modelling community.

6.2 Future work

While our three aims have largely been achieved, there are still numerous avenues of research that could be pursued to improve our findings. The cornerstone of any future work should be on refining our STID map representation and improving on our general docking routine. The work presented in Chapter 4 discussed our current efforts towards this goal. However, it is incomplete; with the general success rate of our proposed methods inferior to the current iteration of JabberDock.

The critical issue we encountered in Section 4.4 was the high frequency of the Potential Energy Surfaces (PES) used for docking, regardless of how smooth the energy profile appeared to be in 1D. This lead to many individual solutions, irrespective of the optimisation method. While an analytically derived scoring function is preferable, some constant is likely necessary to damp the PES oscillations by permitting shape intersection in addition to an alternative optimisation approach. In Section 4.4.5, we discussed methods such as Margin Ranked loss as a means to determine the empirical constant in an adapted scoring function required to return the highest quality models. However, in terms of exploring the space efficiently and mapping local minima which may correspond to the docked pose (as opposed to the global minimum), basin-hopping appears to be the more attractive option. Basin-hopping was designed with complex optimisation problems such as ours in mind,[11] and indeed has been used in rigid protein-protein docking.[12] While it can be expensive,[13] as we showed in Section 4.5, we can rapidly generate starting structures with the surfaces in contact, thereby reducing the overall computational costs. Provided the scoring function is modified to permit intersection, basin-hopping appears to be an attractive solution to following the available gradients while also providing possible correct complexes through mapping high scoring, local minima.

In addition to this larger goal of an improved scoring function and optimisation approach, several, smaller investigations could yield higher-quality structures:

1. Convolve a Lorentzian with the pseudo-atom van der Waals radii as opposed to a Gaussian (see Section 2.2.2). By essentially shrinking the STID map isosurfaces, we may overcome the barriers encountered in Section 4.4 introduced by the necessary intersection during docking. This may facilitate the usage of analytically correct solutions, improving both the speed and quality of results.
2. Use alternative atomistic forcefields to Amber ff14SB[14] in the MD. This will

impact the STID map's topology, if not through the exploration of local conformational space, then by the charges used to build the maps from the forcefield. While it is unknown whether a different choice will improve the quality of the STID isosurfaces, it may be necessary to consider alternative forcefields in unusual systems where no forcefield parameters are available.

3. An additional refinement step to minimise output structures, for example, with MD. Currently, there is no post-refinement step, and while this is acceptable as none of the returned 300 models exhibit side-chain clashes owing to the Lennard-Jones check during optimisation, the prediction returned to the user may yet be an unfeasible high energy structure, despite being close to the ground truth. An MD refinement step would help settle the structure into a more reasonable suggestion.

4. Reduce overall memory/CPU requirements. As we have shown in Section 3.2.2.2, JabberDock is a demanding program to run, particularly on a desktop computer. The dockings performed here were primarily done on various supercomputing clusters. Reducing the computational costs would both enhance the user base, and facilitate further research. Some of the work described in Chapter 4 may go some way towards this goal, but optimising the algorithm is the obvious solution.

5. Leverage on experimental data through integrative modelling to enhance structure prediction, similar to engines like HADDOCK.[15] JabberDock is a blind docking engine, and while as a *de novo* approach, its success rate is highly competitive, there is a clear opportunity to couple our energy function with experimental restraints. Given JabberDock's remarkable improved success rate with the transmembrane protein docking benchmark over the water-soluble set, we can only assume that further constraints will provide even higher-quality models.

6. Dock multimers with a greater number of monomeric units than dimers. JabberDock exclusively targets dimeric proteins. Yet, as we have shown in Section 3.5.1, we can also use the roto-translations returned by JabberDock to provide significantly larger, multimeric structures. Symmetries provide means to predict homomultimeric complexes with memory requirements equivalent to those of a dimer.

7. Upgrade accessibility for users. Many current docking algorithms provide Graphical User Interfaces (GUI) to their userbase. JabberDock is a command-line program, which may disincentivise users with less computational experience, particularly those on a non-Linux based operating system. Providing a webserver to deposit structures, or a GUI for users to run locally, will upgrade the user experience and improve accessibility.

Aside from these individual improvements, there are additional benchmarks to perform that could highlight JabberDock's capability as a universal protein-protein docking engine. For example, using JabberDock's transmembrane docking routine to predict β-barrel formation, although this would require the formulation of a new benchmark set beforehand. Furthermore, we could also apply JabberDock to transmembrane–soluble protein complexes, similar to the recent work by HADDOCK.[16] In these investigations, we envisage that our surrogate bound state work in Section 4.3 could further enhance the quality of returned predictions, although additional work should be performed in this area by benchmarking a much greater range of proteins with bound complex analogues.

In terms of the work discussed in Chapter 5, we foresee that large simulations of the ACE2 receptor could corroborate the various identified oligomeric states of the SARS-CoV-2 spike glycoprotein – ACE2 complexes. Indeed, recent work by Barros *et al.*[17] have highlighted the flexibility of the ACE2 receptor in complex with the receptor-binding domain of the spike. The level of flexibility would certainly confer a greater number of possible oligomeric assemblies, and allow the associated membranes to relax. The MD data from this work was not released at the time of thesis submission, but these questions could be addressed following its release.

Finally, we consider perhaps the greatest challenge facing the protein docking community at present, which unites the general consideration of dynamics in protein structure prediction and continue our discussion from the end of Section 6.1.4. In Chapter 1, we outlined how MD is often used to extract conformational states following allosteric rearrangement. These conformational changes are critical to a protein's function. They typically take place over micro to millisecond timescales, pushing many of them out of reach for Molecular Dynamics for the time being, even with enhanced sampling techniques. Therefore, predicting the conformational dynamics associated with allostery remains a complicated task, and coupling that with predictive protein-protein docking is an even greater challenge. Blindly predicting the quaternary structure of a complex,[18] wherein its monomeric binding units are in a different conformational state prior to binding, is remarkably complicated. Thus, the two fields remain largely segregated. Recent work by Noé *et al.*[19] have utilised deep learning to predict BPTI's conformational states and the free energy barriers between then, paving the way to understanding allosteric regulation without the need for long Molecular Dynamics simulations. Yet, their method is too expensive for larger systems.

The STID map, in its entirety, provides an inviting opportunity to address this. While in this study, we have almost exclusively used an isosurface at a cutoff of 0.43, there is a rich bounty of information stored within the map. Larger values are indicative of more rigid domains, or those that are highly charged, while smaller STID values are associated with highly flexibly or neutral regions. Accessing this diverse set of data is, potentially, the key to considering allosteric conformational dynamics during protein docking. In this, we foresee the use of image analysis techniques borrowed from deep learning. Such methods are typically employed for image classification, but given the similarities between an image's pixels and the STID map's voxels, it seems a natural choice. Therefore, training for docking would require converting unbound STID maps seeded from different conformational states, perhaps sourced through homology modelling or MD, into bound complex maps *via* a generative adversarial network.[20] We would then need to ensure a sufficient backwards mapping routine to reconstruct the atomic structure from this novel STID map. There are significant difficulties to be surmounted for this task, namely that we must place constraints to retain the correct sequence and number of atoms. Furthermore, the size of the images must be consistent without excess zero-padding in the training set, so perhaps this method could only be applied for a specific size/type-subset of proteins at a time. Of course, the long-term goal of this would be to accurately predict complexes formed when only one conformation is available through the PDB; thus, the method would also need to be highly transferable. Regardless, we expect that accessing the full range of STID map data is an avenue of further research to improve protein docking, given the underpinning dynamics hidden therein.

6.3 Concluding remarks

Throughout this thesis, we have endeavoured to demonstrate the importance and viability of harnessing dynamics to elucidate quaternary protein structure prediction, and subsequently, shed light on a protein's function. To this end, we designed the Spatial and Temporal Influence (STID) map, a representation capable of simultaneously representing protein side-chain dynamics, electrostatics and shape. We showed that this representation could represent critical physical and biophysical quantities, and was remarkably successful in a protein-protein docking context. We also demonstrated the usage of large scale Molecular Dynamics simulations to elucidate diverse conformational arrangements between the SARS-CoV-2 spike glycoprotein and ACE2 receptor.

The realisation of our objectives confirms that thorough consideration of a protein's dynamics is imperative in predicting a protein's structure and function. Given this conclusion, it is clear that any future work exploring a protein's function should not rely on models derived from static structures. Instead, they should consider how dynamics can be accounted for in their models or data interpretation. While the dynamics of a protein owes much to its structure, we can not solely rely on the structure to rationalise functionality. Thus, *the structure* ***and dynamics*** *determines the function* of a protein, and it is essential to consider the impact of this in the assembly of proteins.

6.4 Bibliography

[1] W. Wriggers, *Conventions and workflows for using Situs*, *Acta Crystallogr. Sect. D Biol. Crystallogr.*, 2012, **68**, 344–351.

[2] L. Li, C. Li, Z. Zhang and E. Alexov, *On the dielectric "constant" of proteins: Smooth dielectric function for macromolecular modeling and its implementation in DelPhi*, *J. Chem. Theory Comput.*, 2013, **9**, 2126–2136.

[3] H. Fischer, I. Polikarpov and A. F. Craievich, *Average protein density is a molecular-weight-dependent function.*, *Protein Sci.*, 2004, **13**, 2825–8.

[4] A. A. Ashkarran, K. S. Suslick and M. Mahmoudi, *Magnetically Levitated Plasma Proteins*, *Anal. Chem.*, 2020, **92**, 1663–1668.

[5] A. Maißer, V. Premnath, A. Ghosh, T. A. Nguyen, M. Attoui and C. J. Hogan, *Determination of gas phase protein ion densities via ion mobility analysis with charge reduction*, *Phys. Chem. Chem. Phys.*, 2011, **13**, 21630–21641.

[6] T. Vreven, I. H. Moal, A. Vangone, B. G. Pierce, P. L. Kastritis, M. Torchala, R. Chaleil, B. Jiménez-García, P. A. Bates, J. Fernandez-Recio, A. M. J. J. Bonvin and Z. Weng, *Updates to the Integrated Protein-Protein Interaction Benchmarks: Docking Benchmark Version 5 and Affinity Benchmark Version 2*, *J. Mol. Biol.*, 2015, **427**, 3031–3041.

[7] N. Hurwitz, D. Schneidman-Duhovny and H. J. Wolfson, *Memdock: An α-helical membrane protein docking algorithm*, *Bioinformatics*, 2016, **32**, 2444–2450.

[8] S. Viswanath, L. Dominguez, L. S. Foster, J. E. Straub and R. Elber, *Extension of a protein docking algorithm to membranes and applications to amyloid precursor protein dimerization*, *Proteins Struct. Funct. Bioinforma.*, 2015, **83**, 2170–2185.

[9] P. I. Koukos, I. Faro, C. W. van Noort and A. M. Bonvin, *A Membrane Protein Complex Docking Benchmark*, *J. Mol. Biol.*, 2018, **430**, 5246–5256.

[10] L. Casalino, Z. Gaieb, J. A. Goldsmith, C. K. Hjorth, A. C. Dommer, A. M. Harbison, C. A. Fogarty, E. P. Barros, B. C. Taylor, J. S. Mclellan, E. Fadda and R. E. Amaro, *Beyond shielding: The roles of glycans in the SARS-CoV-2 spike protein*, *ACS Cent. Sci.*, 2020, **6**, 1722–1734.

[11] D. J. Wales and J. P. Doye, *Global optimization by basin-hopping and the lowest energy structures of Lennard-Jones clusters containing up to 110 atoms*, *J. Phys. Chem. A*, 1997, **101**, 5111–5116.

[12] B. Olson, I. Hashmi, K. Molloy and A. Shehu, *Basin Hopping as a General and Versatile Optimization Framework for the Characterization of Biological Macromolecules*, *Adv. Artif. Intell.*, 2012, **2012**, 1–19.

[13] M. L. Paleico and J. Behler, *A flexible and adaptive grid algorithm for global optimization utilizing basin hopping Monte Carlo*, *J. Chem. Phys.*, 2020, **152**, 094109.

[14] J. A. Maier, C. Martinez, K. Kasavajhala, L. Wickstrom, K. E. Hauser and C. Simmerling, *ff14SB: Improving the Accuracy of Protein Side Chain and Backbone Parameters from ff99SB*, *J. Chem. Theory Comput.*, 2015, **11**, 3696–3713.

[15] C. Dominguez, R. Boelens and A. M. Bonvin, *HADDOCK: A protein-protein docking approach based on biochemical or biophysical information*, *J. Am. Chem. Soc.*, 2003, **125**, 1731–1737.

[16] J. Roel-Touris, B. Jiménez-García and A. Bonvin, *Integrative Modeling of Membrane-associated Protein Assemblies*, *Nat. Commun.*, 2020, **11**, 6210.

[17] E. P. Barros, L. Casalino, Z. Gaieb, A. C. Dommer, Y. Wang, L. Fallon, L. Raguette, K. Belfon, C. Simmerling and R. E. Amaro, *The Flexibility of ACE2 in the Context of SARS-CoV-2 Infection*, *Biophys. J.*, **accepted**, year.

[18] C. A. Smith, D. Ban, S. Pratihar, K. Giller, M. Paulat, S. Becker, C. Griesinger, D. Lee and B. L. De Groot, *Allosteric switch regulates protein-protein binding through collective motion*, *Proc. Natl. Acad. Sci. U. S. A.*, 2016, **113**, 3269–3274.

[19] F. Noé, S. Olsson, J. Köhler and H. Wu, *Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning*, *Science (80-.).*, 2019, **365**, year.

[20] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, Adv. Neural Inf. Process. Syst., 2014, pp. 2672–2680.