Nancy-Université

École doctorale IAEM Lorraine
UFR STMIA, Université Henri Poincaré

Scientific Memoir

High Performance Algorithms for Molecular Shape Recognition

Presented publicly at 2pm on 5th April 2011 by

David W. Ritchie

for

Authorisation to Manage Research from Université Henri Poincaré

(informatics speciality)

Composition of the jury

Rapporteurs	Gilles Bernot, professeur, Université Nice Sophia Antipolis
	Frederic Cazals, DR INRIA, INRIA Sophia Antipolis – Méditerranée
	Alexandre Varnek, professeur, Université de Strasbourg
Examinateurs	Bernard Girau, professeur, Université Henri Poincaré
	Bruno Lévy, DR INRIA, INRIA Nancy – Grand Est
	Paul Zimmermann, DR INRIA, INRIA Nancy – Grand Est

Laboratoire Lorrain de Recherche en Informatique et ses Applications – UMR 7503

Acknowledgements

“If we knew what we were doing, it would not be called research, would it?” (Albert Einstein).

I didn’t know it at the time, but my research career began about thirteen years ago in Aberdeen when I started studying for a PhD under the supervision of Graham Kemp and John Fothergill. So firstly, I must thank Graham and John for their warmth and enthusiasm during those early years. Subsequently, after I became a lecturer at Aberdeen, teaching and administrative duties didn’t leave a lot of time to do much research of my own. It was therefore always a pleasure to interact with my young (and sometimes not-so-young) research colleagues Alessandra Fano, Antonis Koussounadis, Lazaros Mavridis, Diana Mustard, and Violeta Pérez-Nueno. Much of their projects feature in this memoir, and, although I didn’t know it at the time, much of their work helped to bring me to France. So I would like to thank Alessandra, Antonis, Lazaros, Diana, and Violeta for keeping me closely connected to science, and for being such great people to work with. I would also like to thank several other friends and colleagues at Aberdeen whose company I always enjoyed, and with whom I always liked to interact: Peter Gray, Frank Guerin, Judith Masthof, Chris Mellish, Chris Secombes, and Wamberto Vasconcelos. I will miss Aberdeen!

On the other hand, it is always good to meet new people (and sometimes old friends in new places) and to work on new projects. So I would like to thank Yasmine Asses, Zainab Assaghir, Thomas Bourquard, Matthieu Chavent, Emmanuelle Deschamps, Marie-Dominique Devignes, Léo Ghemtio, Anisah Ghoorah, Stéphane Gégout, Bernard Maigret, Lazaros Mavridis, Amedeo Napoli, Violeta Pérez-Nueno, Malika Smaïl-Tabbone, Michel Souchet, and Vishwesh Venkatraman for helping to make my transition to France such a pleasurable experience. In particular, I am especially grateful to Yasmine Asses, Matthieu Chavent, Anisah Ghoorah, Bernard and Françoise Maigret, and Malika Smaïl-Tabbone for all their help with the French version of this memoir. It would have been impossible without them! Merci à tous et toutes!

Dave Ritchie

Nancy, January 2011.

Contents

Abbreviations

1D:	two-dimensional.
2D:	two-dimensional.
3D:	three-dimensional.
6D:	six-dimensional.
ANR:	Agence Nationale de la Recherche.
AIR:	ambiguous interaction restraint.
AUC:	area under the curve.
CATH:	class, architecture, topology, homology.
CAPRI:	Critical Assessment of PRedicted Interactions.
CoG:	centre of gravity.
CoH:	centre of harmonics.
CPU:	central processing unit.
CUDA:	Common Unified Device Architecture.
DCED:	distance constrained essential dynamics.
DNA:	deoxyribonucleic acid.
ED:	essential dynamics.
EF:	enrichment factor.
EM:	electron microscopy.
ETO:	exponential type orbital.
EVA:	eigenvector analysis.
FG:	Fourier-Gegenbauer.
FFT:	fast Fourier transform.
FN:	false negative.
FP:	false positive.
FPR:	false positive rate.
GL:	Gauss-Laguerre.

GMP: GNU Multiple Precision.
GPU: graphics processing unit.
GTO: Gaussian type orbital.
HIV: human immuno-deficiency virus.
HPC: high performance computing.
HMM: hidden Markov model.
HTVS: high throughput virtual screening.
KDD: knowledge discovery in databases.
LORIA: Laboratoire Lorraine de Recherche en Informatique et ses Applications.
MD: molecular dynamics.
MIF: molecular interaction field.
MLR: mean log rank.
NMA: normal mode analysis.
NMR: nuclear magnetic resonance.
NPC: nuclear pore complex.
PC: personal computer.
PC: physico-chemical.
PC: principal component.
PCA: principal component analysis.
PDB: protein data bank.
PPI: protein-protein interaction.
QM: quantum mechanics.
RDM: relational data mining.
RIF: rotationally invariant fingerprint.
RMS: root mean squared.
RMSD: root mean squared deviation.
RNA: ribonucleic acid.
ROC: receiver-operator characteristic.
ROT: rotational scoring function.
SAS: solvent accessible surface.
SH: spherical harmonic.
SPF: spherical polar Fourier.
TAP-MS: tandem affinity purification by mass spectroscopy.

TN:	true negative.
TP:	true positive.
TPR:	true positive rate.
VS:	virtual screening.
VSM-G:	Virtual Screening Manager – Grids.
VDW:	van der Waals.
Y2H:	yeast two-hybrid.

Chapter 1

Introduction

1.1 Context and Motivation

1.1.1 Molecular Shape Recognition

The main topic of this memoir is the development of computational techniques to represent and compare the three-dimensional structures and properties of molecules. In the context of large biomolecules such as proteins, this involves comparing and classifying their shapes in order to study the relationships between protein structure and function, and it intimately involves predicting how pairs of proteins might fit together or "dock" to form a biomolecular complex. In the context of small organic molecules, it involves comparing and classifying the shapes of candidate drug molecules in order to predict how then might bind to specific protein targets. Therefore, the material in this memoir lies at the interface between computational biology and chemoinformatics.

According to the on-line Wikipedia,[1] computational biology is described as *"... an interdisciplinary field that applies the techniques of computer science, applied mathematics and statistics to address biological problems. The main focus lies on developing mathematical modeling and computational simulation techniques. By these means it addresses scientific research topics with their theoretical and experimental questions without a laboratory..."* On the other hand, the term chemoinformatics was first used by Frank Brown (1998) to describe *"... the use of computing and informatics techniques in chemistry in order to transform data into information and information into knowledge for the intended purpose of making better decisions faster in the area of drug lead identification and optimization."*

Historically, computational biology and chemoinformatics have often been treated as separate disciplines. However, because the three dimensional (3D) structures of both large biomolecules and small drug molecules are central to their function (Bourne & Weissig, 2003; Pevsner, 2003; Petsko & Ringe, 2004), and because drug molecules generally work by modulating the behaviour of their biomolecular targets (Larson, 2006), it is important to consider these two fields together from a sci-

[1] http://en.wikipedia.org/wiki/Computational_biology

entific point of view. Furthermore, I believe that from an informatics point of view it is important to consider all scientific domains equally in order to be able to identify computational or algorithmic techniques which are well known in one domain but which might usefully be extended and transferred to another.

1.1.2 The Importance of Molecular Structure

In biology, it is axiomatic that a protein's amino acid sequence determines its 3D molecular structure, and that a protein's 3D structure determines its specific function. However, due to the enormous volume and considerable complexity of biological data, it is necessary to use advanced computational and visualisation techniques to store and manipulate it. In the current "post-genomic era," in which the entire gene sequences of more and more organisms are being determined routinely, scientific attention is turning towards transforming this basic sequence information into structural and hence functional knowledge. Therefore, the ability to represent and manipulate molecular structures in the computer will become an increasingly important aspect of computational biology. Indeed, computational techniques are already being employed in many areas of the Life Sciences to help make sense of and to exploit the vast quantities of sequence and structural data that is already available. For example, biologists often perform sequence-based alignments and structural superpositions of proteins to gain insights into their biological function, and they may also use computational protein-protein "docking" software to try to predict how pairs of proteins interact at the molecular level. Similarly, medicinal chemists often use protein-ligand docking software to help identify small molecules which might dock onto a given protein target and consequently modulate its behaviour for therapeutic purposes (Jensen, 1999; Dean, 1995; Petsko & Ringe, 2004). Indeed, modeling protein-ligand docking is becoming an increasingly important strategy for structure-based drug discovery (Richards, 2002; Congreve *et al.*, 2005).

1.1.3 Experimental and Computational Bottlenecks

At the experimental level, X-ray crystallography and nuclear magnetic resonance (NMR) are often considered as the "gold standard" techniques for determining high resolution structures of proteins and other macromolecules. However, determining a protein's 3D structure is considerably more difficult than determining its sequence. For example, although some 12,000 distinct protein structures have been deposited in the Protein Data Bank (PDB; Sussman *et al.*., 1998) and new structures are currently being added to the PDB at a rate of over 100 per week, these figures represent only a very small proportion of the total number of known protein sequences. Hence there is an on-going need to be able to create 3D structural models of proteins. Furthermore, only a small proportion of the 3D structures deposited in the PDB correspond to protein-protein complexes, and less than 2% of all known structures comprise protein-protein hetero-complexes. Additionally, due to a number of practi-

cal difficulties it seems unlikely that it will become possible to solve the structures of protein complexes using high-throughput structural genomics techniques in the foreseeable future (Russell *et al.*, 2004). Hence, the use of computational techniques such as homology model-building and protein docking will become increasingly important ways to help understand the molecular mechanisms of biological systems (Aloy *et al.*, 2004; Aloy & Russell, 2006).

When considering molecular structures, it is always important to remember that biomolecules and many small ligand molecules are intrinsically dynamical entities at physiological conditions. For example, the individual atomic positions within a protein rapidly and continuously fluctuate under thermal motion. At longer time-scales, the conformations[2] of the amino acid residues within a protein can flip from one local minimum to another. At even longer time-scales, larger structural subunits such as α-helices and β-sheets may undergo substantial motions which can be very difficult to predict using computational techniques.

Although the fundamental forces between atoms and molecules are almost fully understood at a theoretical level, the practical applications of computer simulation and manipulation in 3D space of large biomolecules such as proteins are often severely limited by the computational costs involved. For example, performing a molecular dynamics (MD) simulation on a single protein domain can involve several days or even weeks of computation. Although some progress is being made, reliably calculating how a pair of proteins interact or "dock" at the molecular level remains a considerable computational challenge. If the flexible nature of protein structures is explicitly taken into account, docking calculations can take up to around 50 CPU-days per complex. Even the task of optimally superposing similar protein structures remains a non-trivial computational problem (Sippl & Wiederstein, 2008). Similarly, structure based drug discovery can be very computationally expensive due to the sheer size of the chemical databases which must be screened. There is therefore a need to develop new computational techniques with which to represent and manipulate the 3D structures of proteins and other molecules, and to simulate protein-protein and protein-ligand interactions in a tractable way.

1.1.4 Protein-Protein Interactions

If DNA represents the biological blueprint for life, then proteins make up the molecular machinery. Proteins often perform their functions by interacting with other proteins to form protein-protein complexes. These complexes may exist as short-lived transitory associations, as in e.g. enzyme catalysis, or as long-lived multimeric systems such as the ribosome, transcription factors, cell surface and ion channel

[2] In chemistry, the term *conformation* is used to describe the relative positions of the atoms within a molecule. Molecules which have the same number, type, and covalent connectivity of atoms may exist in different 3D conformations. A molecule in one conformation can often be transformed into another conformation through rotations about one or more interatomic covalent bonds. There is often a small energy barrier associated with such rotations, but thermal motion often provides sufficient energy to overcome these barriers. Nonetheless, over sufficiently long time scales, proteins typically adopt the conformation with the lowest total energy.

proteins. However, despite knowing the blueprints, we currently know very little about how proteins operate at the molecular level. For example, genome-wide proteomics studies (Uetz *et al.*, 2000; Ito *et al.*, 2001; Gavin *et al.*, 2002; Ho *et al.*, 2002) are providing a growing list of putative protein-protein interactions (PPIs), but understanding the function of these predicted interactions requires further biochemical and structural analysis. For example, yeast is one of the most studied organisms and is known to have around 6,000 proteins, giving rise to between about 38,000 and 75,000 PPIs. Around 50% of these PPIs have been observed experimentally. On the other hand, the human genome encodes around 30,000 proteins, giving from 154,000 to 370,000 PPIs, of which only around 10% are known to date (Aloy & Russell, 2004; Hart *et al.*, 2006). Therefore, developing automated approaches which can "mine" and transfer protein interactions from yeast to human will be an important strategy for populating the human protein interaction network (Bork *et al.*, 2004). Understanding how proteins interact is crucially important for understanding the molecular mechanisms of disease. For example, therapeutic drugs often work by modulating or blocking PPIs, and therefore PPIs represent an important class of drug target (Arkin & Wells, 2004; González-Ruiz & Gohlke, 2006).

1.1.5 Protein Docking

Protein docking is the task of calculating the 3D structure of a protein complex starting from the individual unbound or model-built structures of the constituent proteins. Like all good scientific problems, the protein docking problem is easy to state but hard to solve. Because protein structures are intrinsically dynamic, they can often change conformation on complexation. Hence this computationally intensive task could be likened to trying to assemble the pieces of a complex 3D jigsaw puzzle in which the given parts do not fit together perfectly. In order to make the calculation tractable, most protein docking algorithms begin by assuming that the structures to be docked are rigid. This essentially reduces the problem to a six-dimensional (6D) rotational-translational search space, but it can also cause many incorrect or false-positive docking orientations to be produced. Therefore, the general goal of rigid body docking algorithms is to find a reasonably small number of feasible orientations for a pair of 3D structures which may subsequently be refined and re-scored using more conventional but more computationally intensive techniques. For recent reviews, see Ritchie (2008) and references therein.

In the current state of the art, many protein docking algorithms use rigid-body fast Fourier transform (FFT) correlation techniques to find putative initial docking orientations. This approach was first introduced by Katchalski-Katzir *et al.* (1992) to calculate rapidly shape complementarity within a 3D Cartesian grid, but it was later extended to include additional terms representing electrostatic interactions (Gabb *et al.*, 1997; Mandell *et al.*, 2001), or both electrostatic and desolvation contributions (Chen & Weng, 2003). However, because this approach relies on a Cartesian grid, it is inherently limited to computing translational correlations, and new FFT grids must be computed for each rotational

increment of the rotating molecule. This makes it difficult to incorporate any prior knowledge about a complex in order to focus the calculation around a known or hypothesized binding site. Because fully covering the search space requires many thousands of rotational samples, Cartesian docking algorithms commonly take several hours to complete, and the efficiency of the approach decreases with increasing complexity of the potential. In order to simulate protein flexibility during docking calculations, several groups use FFT techniques to dock *ensembles* of rigid body structures (Grünberg *et al.*, 2004; Mustard & Ritchie, 2005; Smith *et al.*, 2005), and this further increases the computational cost of FFT-based approaches. Due to the increasing use of such multi-term knowledge-based potentials and ensemble docking techniques, there is a growing need to develop more sophisticated and versatile FFT approaches.

1.1.6 Spherical Polar Fourier Docking Correlations

Many current protein docking algorithms use FFTs to calculate *translational* correlations (Eisenstein & Katchalski-Katzir, 2004). In other words, they place the proteins to be docked in a 3D Cartesian grid, and they use an existing 3D FFT library to accelerate the calculation of the translational correlation of one moving protein with respect to the other fixed protein. The FFT part of such docking calculations can be computed rapidly, but it must be repeated for thousands of rotational increments of one of the proteins. Furthermore, many of the translational steps sampled correspond to unrealistic interpenetrations of the two proteins. Additionally, if one bears in mind that two of the rotational degrees of freedom are redundant when the translational component is zero, it follows that many of the orientations sampled in translational FFTs will be almost redundant when such FFTs are applied to multiple rotated starting orientations. In other words, Cartesian-based FFT algorithms evaluate many almost redundant orientations during a 6D correlation search.

The main theme of this memoir is that protein docking and molecular shape recognition tasks are inherently *rotational* problems, and that therefore they should be described using angular coordinate systems so that they can be mapped more naturally to *rotational* FFTs. As indicated above, conventional grid-based FFT docking algorithms typically partition the 6D search space into three rotational and three translational degrees of freedom. However, this allows only three (translational degrees) of freedom to be accelerated by the FFT. On the other hand, a spherical polar coordinate system has one translational and five angular dimensions, and therefore allows the possibility to use FFT techniques to accelerate the calculation in at least five and potentially all six of the degrees of freedom.

My main contribution to the field of protein docking has been to explore and demonstrate the utility of this spherical polar proposition using what I call spherical polar Fourier (SPF) correlations. The basic idea is to represent the shape and electrostatic properties of proteins (or other biomolecules such as DNA and RNA) as high order Fourier expansions of orthonormal spherical harmonic (SH) and Gauss-Laguerre (GL) basis functions. Much of the mathematical theory of this approach is “well

known" in the sense that it derives from the quantum mechanical description of the electronic orbitals of an atom. Nonetheless, I was the first to use this approach to represent the shapes of entire macromolecules, and I showed that this representation is exceptionally well suited for calculating the overlap between pairs of 3D functions (i.e. molecular properties) very rapidly using FFT techniques.

These ideas have been implemented in the *Hex* docking program (Ritchie & Kemp, 2000). During a docking calculation, *Hex* can evaluate many millions of trial orientations per second on an ordinary personal computer (PC). This software has been used successfully in the CAPRI blind docking experiment (Janin *et al.*, 2003; Méndez *et al.*, 2003; Méndez & Wodak, 2005), and is therefore well respected internationally. With over 12,000 internet downloads, *Hex* is used world-wide in academia and industry, and has been cited in over 200 scientific publications.

Very recently, Garzon *et al.* (2009) described a docking correlation approach ("FRODOCK" – fast rotational docking) based on a shape plus electrostatic plus desolvation scoring function in which three rotational degrees of freedom are accelerated by a 3D FFT. FRODOCK is reported to be almost as fast as *Hex*. However, because FRODOCK lacks the special radial functions used in *Hex*, it must use concentric spheres to represent the shapes of proteins and it must perform the 3D translational part of the docking search by explicitly resampling the potential function of the moving partner at a large number of translational samples. Therefore I would claim that my SPF approach continues to define the state of the art in FFT-based protein docking.

1.1.7 Virtual Drug Screening

One of the main activities conducted during the early stages of a drug development project is to use computational techniques to identify or predict small drug-like molecules which might bind to a given protein target and hence change its function. This is often called virtual screening (VS). Structural genomics initiatives are producing protein structures at an increasingly rapid rate, and each new protein to be characterised could serve as a potential new drug target. Therefore, there is a growing and timely opportunity to exploit this emerging structural knowledge for therapeutic purposes. Pharmaceutical companies now have compound databases that contain chemical structure information for literally millions of compounds. Developing improved ways of searching such databases could lead to faster and more cost-effective development of new drug molecules. However, although current protein-ligand docking tools such as Autodock can successfully screen a modest number (i.e. of the order of thousands) of ligands against a given protein target (Park *et al.*, 2006), they are too slow for high throughput VS (HTVS) of corporate chemical databases containing millions of compounds (Khodade *et al.*, 2007). Hence there is a need to develop improved database searching and protein-ligand docking techniques for structure-based drug discovery.

Broadly speaking, there are two main approaches to virtual screening. In receptor-based approaches, the structure of the protein target is known or has been modelled, and the goal is to find

suitable ligands which will e.g. bind near the receptor active site and consequently antagonize (block) the native receptor function. In ligand-based approaches, the structure of the protein target is generally not known, and the goal is to find new ligands which are similar to known antagonists. Some drug molecules act as agonists (i.e. they activate or enhance the native receptor function), but the screening principles are essentially the same as for antagonists. Due to the computational expense of receptor-based (i.e. docking) approaches, HTVS campaigns often employ a combination of approaches, in which ligand-based similarity criteria are used as an initial filter, and candidate compounds that survive this filter are subsequently docked to the protein target. The VSM-G approach nicely exemplifies this filtering principle (Beautrait *et al.*, 2008). If sufficient computational resources are available, it is possible to bypass the ligand-based filter and perform receptor-based screening directly. For example, in the high-profile THINK screen-saver project, over 1,000,000 PCs around the world were made available to provide over 80,000 years of CPU time to screen twelve cancer targets against a virtual ligand database of some 3.5 billion compounds (Davies *et al.*, 2002). However, only the largest corporations and government institutions can afford to employ comparable high performance computing (HPC) resources for HTVS.

1.1.8 Spherical Harmonic Virtual Screening

In my opinion, the current state of the art for efficient 3D molecular shape comparison is based on Gaussian representations of molecular shape (Grant *et al.*, 1996) and the more recent SH surface envelope approach, which was developed independently by myself at Aberdeen (Ritchie & Kemp, 1999; Mavridis *et al.*, 2007), Bernard Maigret at Nancy (Cai *et al.*, 2002; Yamagishi *et al.*, 2006), and Tim Clark at Erlangen (Lin & Clark, 2005). At Aberdeen, my focus was on exploiting the rotational properties of the SH functions to develop a very fast way of superposing and quantitatively comparing the 3D shapes of molecular surfaces. At Nancy, Dr Maigret's work has focused on using SH representations to provide a fast receptor-based filter for virtual screening (Beautrait *et al.*, 2008). At Erlangen, Prof Clark developed the ParaSurf program to represent key quantum mechanical molecular surfaces properties using the SH representation. To complement the Erlangen work, I developed the ParaFit program which can superpose 3D molecular structures calculated by ParaSurf program at rate of up to one hundred molecules per second on a single processor. ParaFit and ParaSurf are currently being marketed by Cepos Insilico Ltd.

1.2 Document Summary and Structure

This document presents a summary of my contributions to the fields of computational molecular shape representation and protein-protein docking using efficient Fourier-like representations of molecular shapes and other properties. As stated above, much of the mathematical theory used here is well

known to theoretical chemists and nuclear physicists, but is generally not well known beyond those specialised fields. Nonetheless, formal mathematical proofs are generally not given here. The interested reader can find such material elsewhere in many excellent reference books on special functions and on quantum mechanics (Talman, 1968; Luke, 1969; Hochstadt, 1971; Biedenharn & Louck, 1981; Sakurai, 1994; Bransden & Joachain, 1997). Instead, the approach taken in the following chapters is to present as axioms any necessary mathematical formulae or results, and to use these as basic building blocks from which to develop computationally efficient 2D and 3D representations and correlation algorithms using only relatively straightforward calculus techniques.

It is worth noting that in the last few years, SH surface representations are increasingly being applied to a broad range of object recognition and registration tasks in areas spanning e.g. anatomy (McPeek *et al.*, 2008; McPeek *et al.*, 2009), civil engineering (Garboczi, 2002; Grigoriu *et al.*, 2006), cryo-electron microscopy (Kovacs & Wriggers, 2002), medical imaging (Frank, 2002b; Edvardson & Smedby, 2003; Huang *et al.*, 2005), and computer graphics and the internet (Kautz *et al.*, 2002; Funkhouser *et al.*, 2003; Kazhdan *et al.*, 2003; Novotni & Klein, 2003; Shen *et al.*, 2009b; Shen *et al.*, 2009a). However, to my knowledge, I remain the only person to have described augmenting the angular SH functions with orthonormal radial functions to construct 3D spherical polar basis functions and to have used this representation successfully to accelerate the calculation of both rotational and translational correlations in 3D space. Therefore, the approaches presented here are novel in the context of protein-protein docking and in 3D shape recognition in general.

The rest of this document is structured as follows. Chapter 2 summarizes some useful mathematical properties of the special functions and demonstrates how they may be used to construct orthogonal basis functions for the representation of 2D and 3D molecular shapes and other properties. Chapter 3 develops this scheme by discussing the action of rotation and translation operators in Hilbert spaces using the Wigner rotation matrices for the spherical harmonics and using a spherical Bessel transform approach to calculate closed form expressions for the corresponding translation matrix elements. Hence these Chapters describe the basic machinery necessary for performing six dimensional Fourier correlations.

Chapter 4 shows how the overall approach may be applied to the task of superposing, comparing, and classifying known protein structures using 3D SPF shape-density representations, and how these representations may be exploited to perform efficient protein docking calculations using 1D, 3D, and 5D rotational FFT correlations. This chapter also describes how a simple 2D SH surface-matching approach may be used to search large chemical databases for high throughput virtual drug screening. Finally, Chapter 5 presents an overview of on-going projects and future objectives. The Appendix lists several peer-reviewed journal articles in which the above work has been published. Unless noted otherwise, all of the molecular graphics figures in this document were produced as screen-shots from my *Hex* program.

Chapter 2

Mathematical Foundations

The main purpose of this chapter is to introduce the mathematical concepts and machinery necessary to represent and manipulate molecular shapes in computationally useful and efficient ways. Much of this material describes the classical special functions of mathematics, which are primarily used here to provide orthonormal basis functions for Fourier-like expansions in 3D space (Luke, 1969; Lebedev, 1972). Because these expansions will involve relatively high order polynomials, it is important to develop efficient methods of calculation which do not sacrifice numerical accuracy. It is also important to use reliable and efficient spatial sampling techniques. Hence this chapter also briefly describes icosahedral tessellation of the sphere.

2.1 The Special Functions

The special functions play an important role in physics and engineering applications because they often appear as the solutions to certain differential and integral equations (Lebedev, 1972). Consequently, they can be used to represent and model many natural phenomena, ranging from electrostatics and electromagnetism to quantum theories of matter. The special functions are smooth analytic polynomials, the terms of which often exhibit certain simple, or “special”, patterns. For example, all of the special functions may be calculated by three-term recursion formulae (Erdélyi *et al.*, 1953a). Similarly, many integrals involving products of special functions often have relatively concise solutions. Before the days of modern computers, these properties greatly facilitated difficult hand calculations. Nowadays, of course, it is possible to programme and calculate much higher order functions than could ever be done by hand. Nonetheless, it is still useful to study the special functions and their properties because using a well-chosen recursion formula or integral transform can lead to dramatic increases in computational performance.

2.1.1 Analytic Functions

In mathematics, analytic functions are smooth infinitely differentiable functions of one or more variables. The most fundamental example of an analytic function is the power series

$$f(x) = \sum_{l=0}^{\infty} a_l x^l, \tag{2.1}$$

where the particular values of the coefficients, a_l, distinguish one function from another. Such a power series may be differentiated any number of times, m, to give

$$\frac{\mathrm{d}^m}{\mathrm{d}x^m} f(x) \equiv f^{(m)}(x) = \sum_{l=0}^{\infty} l(l-1)(l-2)...(l-m+1) a_l x^{l-m}. \tag{2.2}$$

By noting that each coefficient a_l may be isolated by evaluating the l-th derivative of $f(x)$ at $x = 0$,

$$a_l = \frac{f^{(l)}(0)}{l!}, \tag{2.3}$$

the above sum may be written in the form of a Taylor series

$$f(x) = \sum_{l=0}^{\infty} \frac{f^{(l)}(0)}{l!} x^l. \tag{2.4}$$

In other words, if the form of $f(x)$ is already known and is easily differentiable, the coefficients of its power series may be calculated using Eq 2.3. For example, $\cos x$ is given by

$$\begin{aligned} \cos x &= 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + ... \\ &= \sum_{l=0}^{\infty} \frac{(-1)^l x^{2l}}{(2l)!}. \end{aligned} \tag{2.5}$$

These days, we take it for granted that all modern programming languages shall provide built-in routines to calculate standard trigonometric and transcendental functions, typically using power series similar to the above. On the other hand, when the form of $f(x)$ is not known in advance, as is commonly the case in data fitting and analysis problems, it is often not straightforward to isolate and calculate each coefficient. Instead, the coefficients must be calculated "simultaneously" by least-squares techniques, for example, and this can be computationally expensive and error-prone.

2.1.2 Homogeneous 3D Polynomials

Here, the goal is to represent molecular shapes and other molecular properties as high order polynomial expansions in three-dimensional (3D) space. Hence, for example, we seek a representation of the form

$$f(\underline{x}) = \sum_{l=0}^{\infty} (a_l x + b_l y + c_l z)^l, \tag{2.6}$$

with coefficients a_l, b_l, and c_l, and where $\underline{x} = (x, y, z)$ are the usual 3D Cartesian coordinates. Eq 2.6 is sometimes called a solid harmonic function (Hobson, 1931) because one can change to spherical polar coordinates $\underline{r} = (r, \theta, \phi)$ by making the substitutions

$$\begin{aligned} x &= r \sin\theta \cos\phi, \\ y &= r \sin\theta \sin\phi, \\ z &= r \cos\theta, \end{aligned} \tag{2.7}$$

to obtain

$$f(\underline{r}) \equiv f(\underline{x}) = \sum_{l=0}^{\infty} r^l (a_l \sin\theta \cos\phi + b_l \sin\theta \sin\phi + c_l \cos\theta)^l, \tag{2.8}$$

which is clearly a sum of all possible combinations of trigonometric powers, or harmonic frequencies. The relationship between the 3D Cartesian and spherical polar coordinates is shown in Figure 2.1.

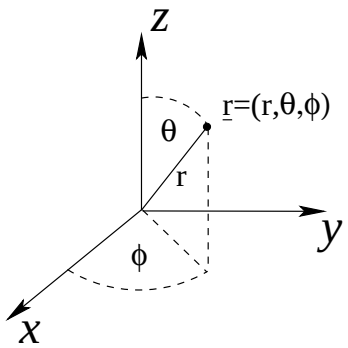

Figure 2.1: The relationship between Cartesian (x, y, z) and spherical polar (r, θ, ϕ) coordinates.

Unfortunately, Equations 2.6 and 2.8 are not very useful for practical purposes because not all of the powers, and hence expansion coefficients, are linearly independent. For example, considering the powers of two, one finds

$$(ax + by + cz)^2 = a^2x^2 + b^2y^2 + c^2z^2 + 2abxy + 2acxz + 2bcyz. \tag{2.9}$$

But from Eq 2.7, it is clear that

$$x^2 + y^2 + z^2 = r^2, \tag{2.10}$$

and therefore x^2, y^2, and z^2 are linearly dependent. Thus, in fact, only five of the six second order Cartesian powers are linearly independent. In a similar manner, it can be shown that only seven of the ten third order harmonic Cartesian powers are linearly independent, and so on. In general, a reliable way to enumerate all of the linearly independent polynomial powers in 3D space is to write

$$f(\underline{r}) = \sum_{l=0}^{\infty} r^l \left(c_l P_{l0}(\cos\theta) + \sum_{m=1}^{l} \left(a_{lm} \cos m\phi + b_{lm} \sin m\phi \right) P_{lm}(\cos\theta) \right), \tag{2.11}$$

where $P_{lm}(\cos\theta)$ are the Legendre polynomials. However, before considering the Legendre polynomials in further detail, it is first worthwhile to review some of the basic properties of the simpler trigonometric, or circular, functions.

2.1.3 The Circular Functions

The trigonometric sine and cosine functions are often called circular functions because they describe the relationship between an angular coordinate ϕ and the Cartesian x and y coordinates along the path of a circle

$$\begin{aligned} x &= \cos\phi, \\ y &= \sin\phi. \end{aligned} \tag{2.12}$$

If a circle of unit radius is drawn at the origin in the complex plane, its coordinates may be represented compactly as a single complex number[1]

$$w = \cos\phi + i\sin\phi, \tag{2.13}$$

where $i = \sqrt{-1}$ is the imaginary unit, and where the real and complex parts of w correspond to the x and y coordinates, respectively. One of the principal results of complex analysis is that the trigonometric functions may be related to the exponential function through Euler's formula

$$e^{i\phi} = \cos\phi + i\sin\phi. \tag{2.14}$$

From this, many further relations follow. For example, De Moivre's formula

$$(\cos\phi + i\sin\phi)^m = \cos m\phi + i\sin m\phi \tag{2.15}$$

follows directly from the fact that

$$(e^{i\phi})^m = e^{i(m\phi)}. \tag{2.16}$$

If one puts $\phi = 2\pi/n$, then the n complex numbers, $e^{i2\pi m/n}$, for $0 \leq m < n$ comprise the vertices of a regular polygon in the complex plane, and are sometimes called the n-th roots of unity. These special points feature prominently in fast Fourier transform (FFT) theory (Kammler, 2000). The use of FFTs in the present work will be described in Chapter 4.

When the circular functions need to be calculated at regular intervals, they may be calculated efficiently using recursion formulae. For example, by writing

$$\cos m\phi = \cos(\phi + (m-1)\phi), \tag{2.17}$$

$$\sin m\phi = \sin(\phi + (m-1)\phi), \tag{2.18}$$

[1] Many text-books use z to represent a complex variable, but I will use w here in order to let x, y, and z correspond to the usual 3D Cartesian coordinates without confusion.

and by applying the identities (which may be derived from De Moivre's formula)

$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta, \tag{2.19}$$

$$\sin(\alpha+\beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta, \tag{2.20}$$

one obtains the stable recursion formulae

$$\cos m\phi = 2\cos\phi\cos(m-1)\phi - \cos(m-2)\phi, \tag{2.21}$$

$$\sin m\phi = 2\cos\phi\sin(m-1)\phi - \sin(m-2)\phi. \tag{2.22}$$

2.1.4 Orthogonal Functions and Hilbert Spaces

The circular functions are orthogonal in the sense that

$$\int_0^{2\pi} e^{im\phi}e^{-ij\phi}\mathrm{d}\phi = 2\pi\delta_{mj}, \tag{2.23}$$

where δ_{mj} is the Kronecker delta

$$\delta_{mj} = \begin{cases} 1 & \text{if } m = j \\ 0 & \text{otherwise.} \end{cases} \tag{2.24}$$

In other words, the total *overlap* between any distinct pair of basis functions is zero. Orthogonal functions play a central role in the theory of Hilbert spaces. A Hilbert space is essentially an algebraic extension of the notion of an ordinary Euclidean space, in which the space is defined by an infinite list of orthogonal basis functions (corresponding to the axes of a Euclidean space) and where any point in that space may be described as a linear combination of basis functions (corresponding to a coordinate vector in Euclidean space) (Debnath & Mikusinski, 1999). One practical application of a Hilbert space is to represent an arbitrary function, $f(\phi)$, in the domain $0 \leq \phi < 2\pi$ as a Fourier series

$$f(\phi) = \sum_{m=0}^{\infty} a_m e^{im\phi}, \tag{2.25}$$

where the functions $e^{im\phi}$ serve as a set of orthogonal basis functions and the expansion coefficients, a_m, serve as "coordinates" with respect to the basis set. Using the orthogonality property, the m-th coefficient may be determined by multiplying each side of Eq 2.25 by $e^{-im\phi}$ and integrating to obtain

$$a_m = \frac{1}{2\pi}\int_0^{2\pi} f(\phi)e^{-im\phi}\mathrm{d}\phi. \tag{2.26}$$

Provided that $f(\phi)$ satisfies some basic conditions about continuity and smoothness (which is normally the case for most physical problems), it can be shown that expansions such as Eq 2.25 converge monotonically in the sense that

$$\lim_{N\to\infty}\left|f(\phi) - \sum_{m=0}^{N} a_m e^{im\phi}\right| = 0. \tag{2.27}$$

Consequently, for practical purposes, one can often represent a complicated function to sufficient accuracy by making a suitable choice for the expansion order, N.

Notationally, it is often useful to use normalised orthogonal, or *orthonormal*, basis functions such as

$$\psi_m(\phi) = \frac{1}{\sqrt{2\pi}} e^{im\phi} \mathrm{d}\phi \tag{2.28}$$

so that the orthogonality property may be written concisely as

$$\int_0^{2\pi} \psi_m(\phi)\psi_j(\phi)^* = \delta_{mj} \tag{2.29}$$

where $\psi_j(\phi)^*$ denotes the complex conjugate of $\psi_j(\phi)$ (i.e. changing i to $-i$). With this convention, the Fourier expansion of some function, $f(\phi)$, becomes

$$f(\phi) = \sum_{m=0}^{\infty} a_m \psi_m(\phi) \tag{2.30}$$

and the expansion coefficients are determined using

$$a_m = \int_0^{2\pi} f(\phi)\psi_m(\phi)^* \mathrm{d}\phi. \tag{2.31}$$

Algebraically, the action of multiplying both sides by some function and integrating is often called an *integral transform.* When working with Hilbert spaces or special functions (Debnath & Mikusinski, 1999), it is often the case that performing a suitable integral transform is a useful way to proceed (Debnath & Bhatta, 2007).

2.1.5 The Gamma Function and Related Factorials

Euler's Gamma function, $\Gamma(w)$, may be defined by the integral

$$\int_0^{\infty} e^{-t} t^w \mathrm{d}t = \Gamma(w+1). \tag{2.32}$$

The Gamma function may be considered as a generalised factorial function in the sense that

$$\begin{aligned} \Gamma(w+1) &= w\Gamma(w) \\ &= w(w-1)\Gamma(w-1) \\ &= w(w-1)(w-2)\Gamma(w-2) \\ &\text{... etc.} \end{aligned} \tag{2.33}$$

When $w = 0$, it can be seen from Eq 2.32 that $\Gamma(0) = 1$. When w is a real integer, the Gamma function reduces to an ordinary factorial

$$\begin{aligned} \Gamma(n+1) &= n(n-1)(n-2)...\Gamma(0) \\ &= n! \end{aligned} \tag{2.34}$$

For the special case of $w = 1/2$, Eq 2.32 may be used again to show that

$$\Gamma(1/2) = \sqrt{\pi}. \tag{2.35}$$

For other values of w, the Gamma function may be estimated rather accurately using Lanczos' formula (Lanczos, 1964). Here, we are mostly concerned with evaluating Gamma functions or products of Gamma functions of integer or half-integer argument up to around $w = 128$. Hence, some care is required in order to avoid numerical overflow and to preserve high arithmetic precision. It is therefore convenient to introduce some further notation which will facilitate cancelling common factors in complex expressions. Specifically, the falling factorial, $[w]_k$, may be defined as

$$[w]_k = w(w-1)(w-2)...(w-k+1). \tag{2.36}$$

Similarly, the rising factorial, $(w)_k$, may be defined as

$$(w)_k = w(w+1)(w+2)...(w+k-1). \tag{2.37}$$

For integer arguments, these factorials become

$$[n]_k = n(n-1)(n-2)...(n-k+1) = \frac{n!}{(n-k)!} \tag{2.38}$$

and

$$(n)_k = n(n+1)(n+2)...(n+k-1) = \frac{(n+k-1)!}{(n-1)!}. \tag{2.39}$$

In particular, $\Gamma(n+1/2)$ may be calculated using the identity

$$\Gamma(n+1/2) = \sqrt{\pi}(1/2)_n \tag{2.40}$$

and the fact that

$$\begin{aligned}(1/2)_n &= \left(\frac{1}{2}\right)\left(\frac{3}{2}\right)\left(\frac{5}{2}\right)...\left(\frac{2n-1}{2}\right) \\ &= (1.3.5.7...(2n-1))\left(\frac{1}{2}\right)^n.\end{aligned} \tag{2.41}$$

Sometimes it is also useful to define a general binomial coefficient as

$$\binom{\alpha}{m} = \frac{\Gamma(\alpha+1)}{\Gamma(\alpha+1-m)m!}. \tag{2.42}$$

2.1.6 Symbolic Simplification of Factorials

Several of the expressions used here involve multiplication and division of relatively high order factorials. However, because these expressions often contain many terms which can be cancelled, it is useful to implement a symbolic method of simplifying them as far as possible before performing any arithmetic. For example, a binomial coefficient may be calculated as

$$\binom{n}{m} = \frac{n!}{(n-m)!m!} = \frac{(0^1 \times 1^1 \times 2^1 ... n^1)}{(0^1 \times 1^1 \times 2^1 ... (n-m)^1)(0^1 \times 1^1 \times 2^1 ... m^1)}. \tag{2.43}$$

Clearly, arbitrary combinations of factorials can be accumulated by adding and subtracting integer powers of the factors in the numerator and denominator, respectively. For example,

$$\binom{6}{2} = 0^{-1} \times 1^{-1} \times 2^{-1} \times 3^0 \times 4^0 \times 5^1 \times 6^1 = \frac{5 \times 6}{2}. \tag{2.44}$$

Furthermore, any such expression can be symbolically reduced further to products of powers of prime numbers. For example,

$$\frac{5 \times 6}{2} = 2^0 \times 3^1 \times 5^1. \tag{2.45}$$

A small set of utility functions has been implemented in C to carry out such manipulations automatically for the above integer factorials and powers, and to perform any remaining arithmetic using the GMP high precision mathematical library.[2] For example, using these utilities an expression such as

$$x = \frac{n!}{\Gamma(n + 1/2)} \tag{2.46}$$

can be evaluated using the fragment of C code shown in Figure 2.2.

```
#include <math.h>
double x;
sp_init();                          // initialise working memory
sp_fac(n, +1);                      // set numerator = n!
sp_rise2(n, -1);                    // set divisor   = (1/2)_n
x = sp_ans() / sqrt(M_PI);          // simplify and supply result
```

Figure 2.2: C programming example to illustrate symbolic simplification and evaluation of expressions involving products of factorials.

[2] http://gmplib.org/.

2.1.7 The Jacobi Polynomials

The Jacobi polynomials, $P_k^{(\alpha,\beta)}(x)$, may be defined as (Erdélyi *et al.*, 1953a)

$$P_k^{(\alpha,\beta)}(x) = \frac{1}{2^k}\sum_{j=0}^{k}\binom{k+\alpha}{j}\binom{k+\beta}{k-j}(x+1)^j(x-1)^{k-j}. \tag{2.47}$$

In some applications, it is convenient to use the shifted expansion (Keister & Polyzou, 1997)

$$P_k^{(\alpha,\beta)}(x) = \frac{\Gamma(k+\alpha+1)}{k!\Gamma(k+\lambda)}\sum_{j=0}^{k}(-1)^j\binom{k}{j}\frac{\Gamma(k+j+\lambda)}{\Gamma(j+\alpha+1)}\left(\frac{1-x}{2}\right)^j, \tag{2.48}$$

where $\lambda = \alpha+\beta+1$. The inverse expansion is given by (Erdélyi *et al.*, 1953b)

$$(1-x)^k = 2^k k!\Gamma(k+\alpha+1)\sum_{j=0}^{k}(-1)^j\frac{(2j+\lambda)\Gamma(j+\lambda)}{(k-j)!\Gamma(k+j+\lambda+1)\Gamma(j+\alpha+1)}P_j^{(\alpha,\beta)}(x). \tag{2.49}$$

The Jacobi polynomials are orthogonal with respect to a weight factor $(1-x)^\alpha(1+x)^\beta$ in the sense that

$$\int_{-1}^{1}(1-x)^\alpha(1+x)^\beta P_k^{(\alpha,\beta)}(x)P_n^{(\alpha,\beta)}(x)\mathrm{d}x = \frac{\Gamma(k+\alpha+1)\Gamma(k+\beta+1)}{k!\Gamma(k+\lambda)}\frac{2^\lambda}{2k+\lambda}\delta_{kn}. \tag{2.50}$$

When $k \geq 2$, the Jacobi polynomials may be calculated *via* the stable recursion formula

$$\begin{aligned}2(k+1)(k+\alpha+\beta+1)(2k+\alpha+\beta)P_{k+1}^{(\alpha,\beta)}(x) = \\ (2k+\alpha+\beta+1)\big[(2k+\alpha+\beta)(2k+\alpha+\beta+2)x+\alpha^2-\beta^2\big]P_k^{(\alpha,\beta)}(x) - \\ 2(k+\alpha)(k+\beta)(2k+\alpha+\beta+2)P_{k-1}^{(\alpha,\beta)}(x).\end{aligned} \tag{2.51}$$

The Jacobi polynomials are the most general type of orthogonal function in the interval $[-1,1]$. The Gegenbauer (or ultra-spherical) polynomials arise when $\alpha = \beta$. The Legendre polynomials appear when $\alpha = \beta = 0$, and the Chebychev polynomials correspond to the special case of $\alpha = \beta = -1/2$.

2.1.8 The Legendre Polynomials

The Legendre polynomials, $P_l(\mu)$, of order $l \geq 0$ may be defined using Rodrigues' formula

$$P_l(\mu) = \frac{1}{2^l l!}\frac{\mathrm{d}^l}{\mathrm{d}\mu^l}(\mu^2-1)^l. \tag{2.52}$$

More generally, the associated Legendre polynomials, $P_{lm}(\mu)$, of order l and degree $m \leq l$, may be defined as

$$P_{lm}(\mu) = (-1)^m\frac{(1-\mu^2)^{m/2}}{2^l l!}\frac{\mathrm{d}^{l+m}}{\mathrm{d}\mu^{l+m}}(\mu^2-1)^l. \tag{2.53}$$

From now on, the term “Legendre polynomial” will be taken to mean the general polynomial, $P_{lm}(\mu)$, where $|m| \leq l$. Legendre polynomials in which m is negative may be calculated using the identity (Hobson, 1931)

$$P_{l\overline{m}}(\mu) = (-1)^m P_{l|m|}(\mu). \tag{2.54}$$

The natural domain of the Legendre polynomials is $-1 \leq \mu < 1$, or with the change of variable, $\mu = \cos\theta$, the domain becomes $0 \leq \theta < \pi$. Thus these polynomials may equally be defined as

$$P_{lm}(\theta) = (-1)^m \frac{(\sin\theta)^m}{2^l l!} \frac{\mathrm{d}^{l+m}}{\mathrm{d}\mu^{l+m}} (\cos\theta)^l. \tag{2.55}$$

The Legendre polynomials are orthogonal in the sense that

$$\int_{-1}^{1} P_{km}(\mu) P_{lm}(\mu) \mathrm{d}\mu = \frac{2}{(2l+1)} \frac{(l+m)!}{(l-m)!} \delta_{kl}. \tag{2.56}$$

An explicit power series expression for the Legendre polynomials may be obtained by expanding $(\mu^2 - 1)^l$ as a binomial series and by differentiating it $l + m$ times:

$$P_{lm}(\mu) = (1-\mu^2)^{m/2} \sum_{k=\frac{l+m+1}{2}}^{l} \frac{(-1)^{k+l+m}}{2^l l!} \binom{l}{k} \frac{(2k)!}{(2k-l-m)!} \mu^{2k-l-m}, \tag{2.57}$$

where the lower summation bound is taken using integer truncation. Often, the Legendre polynomials are calculated using recursion relations. For example, the conventional recursion formula, modified to include the Condon-Shortley phase factor (Condon & Odabasi, 1980), starts with

$$P_{ll}(\theta) = (-1)^l \frac{(2l)!}{l!} \left(\frac{\sin\theta}{2} \right)^l \tag{2.58}$$

and continues down to $m = 0$ with (Hobson, 1931)

$$P_{lm}(\theta) = -2(m+1)\cot(\theta) P_{l,m+1}(\theta) - \frac{P_{l,m+2}(\theta)}{(l-m)(l+m+1)}, \tag{2.59}$$

where $P_{lm}(\theta) = 0$ when $m > l$. This has very good numerical stability (Wiggins & Saito, 1971; Libbrecht, 1985).

2.1.9 The Spherical Harmonics

The regular solid SH functions, $Y_{lm}(\underline{r})$, are usually expressed as complex functions of the spherical polar coordinates (Hobson, 1931):

$$Y_{lm}(\underline{r}) = r^l Y_{lm}(\theta, \phi), \tag{2.60}$$

where $-l \leq m \leq +l$. Here, we are mostly concerned with the surface harmonic functions, $Y_{lm}(\theta, \phi)$, sometimes called tesseral harmonics, obtained by setting $r = 1$. The SHs are separable

$$Y_{lm}(\theta, \phi) = \vartheta_{lm}(\theta)\psi_m(\phi), \tag{2.61}$$

where $\vartheta_{lm}(\theta)$ are normalised Legendre polynomials

$$\vartheta_{lm}(\theta) = \left[\frac{(2l+1)}{2}\frac{(l-m)!}{(l+m)!}\right]^{1/2} P_{lm}(\cos\theta), \tag{2.62}$$

and where $\psi_m(\phi)$ are normalised circular functions

$$\psi_m(\phi) = \frac{1}{\sqrt{2\pi}} e^{im\phi}. \tag{2.63}$$

Thanks to the orthogonality of the circular and Legendre functions, the SHs are orthonormal in the sense that

$$\int_0^{2\pi}\int_0^{\pi} Y_{lm}(\theta, \phi)Y_{l'm'}(\theta, \phi)^* \sin\theta \mathrm{d}\theta\mathrm{d}\phi = \delta_{ll'}\delta_{mm'}. \tag{2.64}$$

The SHs are important in many areas of physics because they appear as solutions to Laplace's equation. Many natural phenomena such as e.g. gravitation, electrostatics, and fluid and heat flow may be described by conservative potentials (i.e. those that depend only on the spatial position of a particle). Laplace's equation states that for a conservative potential, $\psi(\underline{r})$:

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)\psi(\underline{r}) = 0. \tag{2.65}$$

This is often written using the "Laplacian" operator, ∇^2,

$$\nabla^2\psi(\underline{r}) = 0. \tag{2.66}$$

Using standard calculus techniques to change variables, the Laplacian may be written in polar coordinates as:

$$\nabla^2 = \frac{1}{r}\left(\frac{\partial^2}{\partial r^2}\right)r + \frac{1}{r^2}\Lambda^2 \tag{2.67}$$

where Λ^2 is the "Legendrian" operator:

$$\Lambda^2 = \frac{1}{\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial}{\partial\theta}\right) + \frac{1}{\sin^2\theta}\frac{\partial^2}{\partial\phi^2}. \tag{2.68}$$

The angular part of Laplace's equation has solutions:

$$\Lambda^2 Y_{lm}(\theta, \phi) = -l(l+1)Y_{lm}(\theta, \phi). \tag{2.69}$$

In other words, the SHs are eigenfunctions of the Legendrian operator. The radial part of Laplace's equation is then found to admit two solutions of the form r^l and $r^{-(l+1)}$ which correspond to the regular and irregular solid harmonics, respectively.

The regular solid SHs may be written in terms of Cartesian coordinates by substituting the identity

$$z^m = (r\cos\theta)^m \tag{2.70}$$

into the power series definition of the normalised Legendre polynomials, Eq.s 2.56 and 2.57, and similarly by noting that

$$\begin{aligned}(x+iy)^m &= (r\sin\theta(\cos\phi + i\sin\phi))^m \\ &= (r\sin\theta)^m e^{im\phi},\end{aligned} \tag{2.71}$$

to give

$$Y_{lm}(\underline{x}) = \left[\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}\right]^{1/2} \sum_{k=\frac{l+m+1}{2}}^{l} \binom{l}{k} \frac{(-1)^{k+l+m}(2k)!}{2^l l!(2k-l-m)!} \frac{(x+iy)^m z^{2k-l-m}}{r^{2k}}. \tag{2.72}$$

2.1.10 Spherical Harmonic Coupling Coefficients

When calculating certain integrals involving SH functions, it is often useful to expand a product of SHs as a linear combination (Biedenharn & Louck, 1981)

$$Y_{lm}(\theta,\phi)Y_{l'm'}(\theta,\phi) = \sum_{kj}\left[\frac{(2l+1)(2l'+1)}{4\pi(2k+1)}\right]^{1/2} C^{l\,l'\,k}_{000} C^{l\,l'\,k}_{mm'\,j} Y_{kj}(\theta,\phi), \tag{2.73}$$

where $C^{l\,l'\,k}_{mm'\,j}$ is the Clebsh-Gordan coupling coefficient. Wigner's formula for the Clebsh-Gordan coefficient is given by

$$\begin{aligned}C^{j_1\,j_2\,j}_{m_1\,m_2\,m} = \delta_{m_1+m_2,m}&\left[(2j+1)\frac{(j+j_1-j_2)!(j-j_1+j_2)!(j_1+j_2-j)!}{(j_1+j_2+j+1)!}\right]^{1/2}\times \\ &\left[\frac{(j+m)!(j-m)!}{(j_1+m_1)!(j_1-m_1)!(j_2+m_2)!(j_2-m_2)!}\right]^{1/2}\times \\ &\sum_k (-1)^{j_2+m_2+k}\frac{(j_2+j+m_1-k)!(j_1-m_1+k)!}{(j-j_1+j_2-k)!(j+m-k)!(j_1-j_2-m+k)!k!},\end{aligned} \tag{2.74}$$

where the summation is over all values of k for which the factorials are well defined. When $m_1 = m_2 = m = 0$, the coupling coefficient vanishes unless $j_1 + j_2 + j$ is even, in which case the expression can be reduced to

$$C^{j_1\,j_2\,j}_{0\,0\,0} = \left[(2j+1)\frac{(j_1+j_2-j)!(j_1-j_2+j)!(j_2-j_1+j)!}{(j_1+j_2+j+1)!}\right]^{1/2}\frac{(-1)^{l-j}l!}{(l-j_1)!(l-j_2)!(l-j)!}, \tag{2.75}$$

where $l = (j_1 + j_2 + j)/2$. From the permutational symmetries of the Clebsch-Gordan coefficients,

$$\begin{aligned} C^{j_1\ j_2\ j}_{m_1\, m_2\, m} &= (-1)^{j_1+j_2-j} C^{j_2\ j_1\ j}_{m_2\, m_1\, m} \\ &= (-1)^{j_1+j_2-j} C^{j_1\ j_2\ j}_{\overline{m}_1\, \overline{m}_2\, \overline{m}}, \end{aligned} \tag{2.76}$$

it follows that

$$C^{j_1\ j_2\ j}_{m\, \overline{m}\, 0} = (-1)^{j_1-j_2-j} C^{j_1\ j_2\ j}_{\overline{m}\, m\, 0} \tag{2.77}$$

and

$$C^{j_1\ j_2\ j}_{m\, \overline{m}\, 0} = C^{j_2\ j_1\ j}_{m\, \overline{m}\, 0}. \tag{2.78}$$

Wigner's *3-j* symbol, which will appear in the next chapter, is closely related to the Clebsch-Gordan coefficients, and is given by

$$\begin{pmatrix} j_1 & j_2 & j \\ m_1 & m_2 & \overline{m} \end{pmatrix} = \frac{(-1)^{m+j_1-j_2}}{\sqrt{2j+1}} C^{j_1\ j_2\ j}_{m_1\, m_2\, m}. \tag{2.79}$$

High order *3-j* symbols may be calculated efficiently by recursion, and by using Wigner's formula only to seed the recursion. For example, the recurrence relations of Sakurai (1994) for fixed j_1, j_2, and j are

$$\begin{aligned} \big[(j+m)(j-m+1)\big]^{1/2} C^{j_1\ j_2\ j}_{m_1,m_2,m-1} =& \big[(j_1+m_1+1)(j_1-m_1)\big]^{1/2} C^{j_1\ j_2\ j}_{m_1+1,m_2,m} \\ &- \big[(j_2+m_2+1)(j_2-m_2)\big]^{1/2} C^{j_1\ j_2\ j}_{m_1,m_2+1,m} \end{aligned} \tag{2.80}$$

and

$$\begin{aligned} \big[(j-m)(j+m+1)\big]^{1/2} C^{j_1\ j_2\ j}_{m_1,m_2,m+1} =& \big[(j_1-m_1+1)(j_1+m_1)\big]^{1/2} C^{j_1\ j_2\ j}_{m_1-1,m_2,m} \\ &- \big[(j_2-m_2+1)(j_2+m_2)\big]^{1/2} C^{j_1\ j_2\ j}_{m_1,m_2-1,m}. \end{aligned} \tag{2.81}$$

The required coefficients $C^{j_1\ j_2\ j}_{m\, \overline{m}\, 0}$ can be calculated by evaluating Eqs 2.80 and 2.81 alternately in a "zig-zag" path in the m_1/m_2 plane (Sakurai, 1994). Since these recursion formulae are not especially stable, numerical errors can be reduced by using both upwards recursion from $C^{j_1\ j_2\ j}_{000}$, and downwards recursion from $C^{j_1\ j_2\ j}_{m,\overline{m},0}$. The accuracy of the calculation can be assessed from the various orthogonality relations of the Clebsch-Gordan coefficients (Biedenharn & Louck, 1981). Although a detailed numerical analysis has not been performed, I find that rounding errors become significant once j_1 and j_2 reach about 20, even when using quadruple precision hardware arithmetic. For large quantum numbers, a successful approach is to use Wigner's formula (Eq 2.74) and to reduce all factorials symbolically to products of primes (see Section 2.1.6) prior to completing the calculation in 256-bit arithmetic using the GMP library.

2.1.11 Real Spherical Harmonics

The real SHs, $y_{lm}(\theta,\phi)$, may be found by taking linear combinations of the complex functions

$$y_{lm}(\theta,\phi) = \begin{cases} (Y_{lm}(\theta,\phi) + Y_{lm}(\theta,\phi)^*)/\sqrt{2} & \text{if } m > 0 \\ Y_{l0}(\theta,\phi) & \text{if } m = 0 \\ -i(Y_{l\overline{m}}(\theta,\phi) - Y_{l\overline{m}}(\theta,\phi)^*)/\sqrt{2} & \text{if } m < 0. \end{cases} \tag{2.82}$$

Hence the real SHs are also eigenfunctions of Laplace's equation. Expanding the various terms and using Eq 2.54 gives

$$y_{lm}(\theta,\phi) = \begin{cases} \vartheta_{lm}(\theta)(\cos m\phi)/\sqrt{\pi} & \text{if } m > 0 \\ \vartheta_{lm}(\theta)/\sqrt{2\pi} & \text{if } m = 0 \\ \vartheta_{l\overline{m}}(\theta)(\sin \overline{m}\phi)/\sqrt{\pi} & \text{if } m < 0, \text{ i.e. } \overline{m} > 0. \end{cases} \tag{2.83}$$

Thus the real SH functions may be written as

$$y_{lm}(\theta,\phi) = \vartheta_{l|m|}(\theta)\varphi_m(\phi) \tag{2.84}$$

where

$$\varphi_m(\phi) = \begin{cases} \cos m\phi/\sqrt{\pi} & \text{if } m > 0 \\ 1/\sqrt{2\pi} & \text{if } m = 0 \\ \sin \overline{m}\phi/\sqrt{\pi} & \text{if } m < 0. \end{cases} \tag{2.85}$$

Figure 2.3 shows the shapes of the real SHs up to l=2.

It is often considerably more efficient to represent real quantities using real SHs because this allows all complex arithmetic to be avoided. However, as shown in Chapter 4, when using FFTs to accelerate calculations, it is also useful to be able to switch between complex and real bases. Hence it is useful to write linear combinations such as Eq 2.82 in matrix form as

$$y_{lm}(\theta,\phi) = \sum_{m'=-l}^{l} U^{(l)}_{mm'} Y_{lm}(\theta,\phi), \tag{2.86}$$

where $U^{(l)}$ is a unitary matrix, i.e. the complex conjugate transpose of $U^{(l)}$ is the inverse matrix (Biedenharn & Louck, 1981). Noting that all the off-diagonal elements of $U^{(l)}$ are zero, the change of basis equations can be shown more explicitly in matrix form as

$$\begin{pmatrix} y_{ll} \\ y_{lm} \\ y_{l0} \\ y_{l\overline{m}} \\ y_{l\bar{l}} \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}} & 0 & 0 & 0 & \frac{(-1)^l}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{(-1)^m}{\sqrt{2}} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{i(-1)^m}{\sqrt{2}} & 0 & \frac{-i}{\sqrt{2}} & 0 \\ \frac{i(-1)^l}{\sqrt{2}} & 0 & 0 & 0 & \frac{-i}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} Y_{ll} \\ Y_{lm} \\ Y_{l0} \\ Y_{l\overline{m}} \\ Y_{l\bar{l}} \end{pmatrix}. \tag{2.87}$$

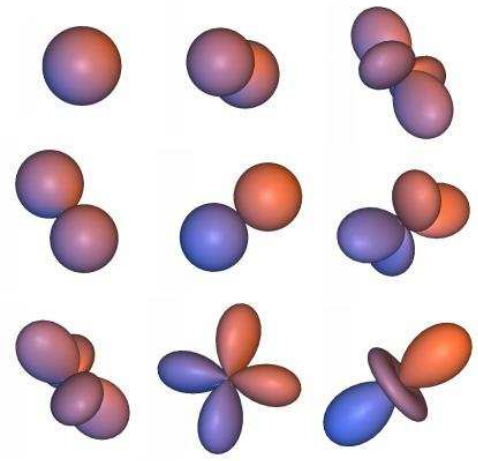

Figure 2.3: The shapes of the real SH functions up to order l=2. Starting from the sphere, y_{00}, at the top left of this figure, the adjacent three functions have l=1, and the five functions in the final row and column have l=2. The three functions on the main diagonal have m=0; functions with positive values of m are shown above the main diagonal, and those with negative m indices are shown below the main diagonal.

2.1.12 The Laguerre Polynomials

The generalised Laguerre polynomials, $L_k^{(\alpha)}(t)$, may be defined by Rodrigue's formula

$$L_k^{(\alpha)}(t) = \frac{t^{-\alpha}e^t}{k!}\frac{\mathrm{d}^k}{\mathrm{d}t^k}\left(e^{-t}e^{k+\alpha}\right). \tag{2.88}$$

An equivalent binomial expansion is given by (Erdélyi *et al.*, 1953b)

$$L_k^{(\alpha)}(t) = \sum_{j=0}^{k}\binom{k+\alpha}{k-j}\frac{(-t)^j}{j!}. \tag{2.89}$$

The Laguerre polynomials have positive real roots within the interval $[0, k+\alpha+(k-1)\sqrt{(k+\alpha)}]$. High order polynomials may be calculated efficiently using the stable recursion

$$(k+1)L_{k+1}^{(\alpha)}(t) = (2k+\alpha+1-t)L_k^{(\alpha)}(t) - (k+\alpha)L_{k-1}^{(\alpha)}(t), \tag{2.90}$$

along with the identities

$$L_0^{(\alpha)}(t) = 1 \tag{2.91}$$

and

$$L_1^{(\alpha)}(t) = \alpha + 1 - t. \tag{2.92}$$

The Laguerre polynomials are orthogonal with respect to a weight factor, $e^{-t}t^{\alpha}$, in the sense that

$$\int_0^\infty e^{-t}t^{\alpha}L_k^{(\alpha)}(t)L_{k'}^{(\alpha)}(t)\mathrm{d}t = \frac{\Gamma(k+\alpha+1)}{k!}\delta_{kk'}. \tag{2.93}$$

2.1.13 GTO and ETO Radial Basis Functions

Although the solid harmonics are natural candidates for representing smooth conservative potentials, in order to be able to represent arbitrary 3D molecular shapes it would be desirable to use radial functions such as the Laguerre polynomials which exhibit radial nodes or zeros similar to the angular zeros of the SHs. Taking into account the orthogonality condition for the Laguerre polynomials in the previous section, it therefore seems reasonable to consider radial functions of the form:

$$R_k^{(\alpha)}(r) = N_k^{(\alpha)} e^{-t/2} t^{\alpha/2} L_k^{(\alpha)}(t) \tag{2.94}$$

where $N_k^{(\alpha)}$ is a normalisation factor and where t is a suitably scaled radial distance. Because the 3D volume element is given by

$$\mathrm{d}V = r^2 \mathrm{d}r \sin\theta \mathrm{d}\theta \mathrm{d}\phi, \tag{2.95}$$

the manner in which the radial distance r is scaled onto the formal parameter t will then fix the value of α. Specifically, by choosing a scale factor λ and making a change of variable

$$t = r^2/\lambda \tag{2.96}$$

one obtains

$$r^2 \mathrm{d}r = \frac{\lambda^{3/2}}{2} t^{1/2} \mathrm{d}t, \tag{2.97}$$

and this entails putting $\alpha = l + 1/2$ in order to maintain the form of the orthogonality relation, Eq 2.93. Because these functions will be used together with the SHs, the assignment $k = n - l - 1$ is made below to ensure that the products of such radial functions with the SHs will enumerate distinct combinations of products of powers of $x = r\sin\theta\cos\phi$, $y = r\sin\theta\sin\phi$, and $z = r\cos\theta$. Without giving a formal proof, this will ensure that the final set of orthogonal basis functions will be *complete*. Some further working then gives

$$R_{nl}(r) = \left[\frac{2}{\lambda^{3/2}} \frac{(n-l-1)!}{\Gamma(n+1/2)}\right]^{1/2} e^{-\rho^2/2} \rho^l L_{n-l-1}^{(l+1/2)}(\rho^2), \tag{2.98}$$

where now $\rho^2 = r^2/\lambda$.

These Gauss-Laguerre (GL) functions are orthonormal in the sense that

$$\int_0^\infty R_{nl}(r) R_{n'l}(r) r^2 \mathrm{d}r = \delta_{nn'}. \tag{2.99}$$

Because these functions have a Gaussian pre-factor, 3D GL plus SH basis functions are often called Gaussian-type orbitals (GTOs) in the quantum chemistry literature. The Gaussian factor ensures that these functions tend to zero for large radial parameter values. Indeed, from Section 2.1.12, the zeros of these functions are bounded within the range $[0, n - 1/2 + (n-l-2)\sqrt{(n-1/2)}]$. When docking

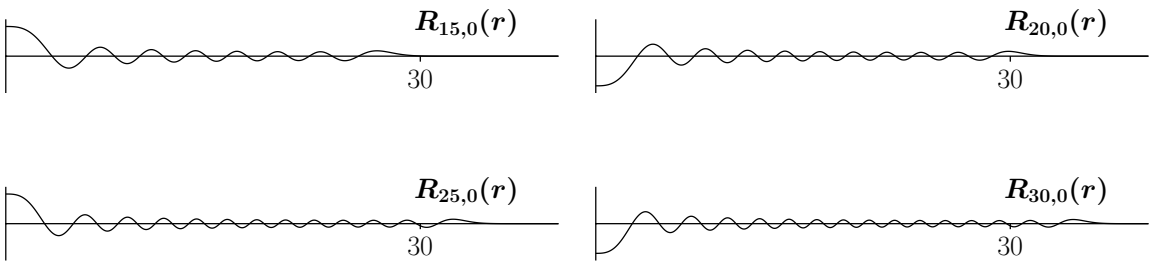

Figure 2.4: The shapes of some example GTO radial basis functions, scaled to 30Å.

typical protein domains, setting a scale factor of $\lambda = 20$ gives good results for globular domains with an average radius of up to around 30Å (Ritchie, 1998). Figure 2.4 shows the shapes of some GTO basis functions.

It is also possible to make a linear change of variable

$$t = \rho = \Lambda r, \tag{2.100}$$

with scale factor Λ. This entails setting $\alpha = 2l + 2$ to satisfy the orthogonality condition. Some further working then gives the orthonormal functions

$$S_{nl}(r) = \left[(2\Lambda)^3 \frac{(n-l-1)!}{(n+l+1)!}\right]^{1/2} e^{-\rho/2} \rho^l L_{n-l-1}^{(2l+2)}(\rho). \tag{2.101}$$

In quantum chemistry, these radial functions are often called exponential-type orbitals (ETOs). I use ETOs to represent the electrostatic properties of proteins.

It is worth pointing out that when I started looking for suitable radial functions during my PhD thesis (Ritchie, 1998), I was inspired by Schrödinger's equation for the hydrogen atom which has the form

$$\psi_{nlm}(\underline{r}) = N_{nl} e^{-\rho/2} \rho^l L_{n-l-1}^{(2l+1)}(\rho) Y_{lm}(\theta, \phi), \tag{2.102}$$

where N_{nl} is a normalisation factor and ρ is a scaled distance. In Schrödinger's equation, the index n corresponds to the principal quantum number of the hydrogen atom which by convention counts from unity. I followed the same numbering convention in my early publications. This tends to make some of the formulae below slightly more verbose than necessary, but I continue to using this convention for consistency.

2.1.14 The Bessel Functions

Finally, the Bessel functions are introduced here because they provide an analytic way to describe the relationship between different coordinate frames in polar coordinates. In other words, they provide

the analytic machinery with which to calculate translations of polar Fourier expansions. The general Bessel function, $J_\nu(w)$, of degree ν and complex argument w may be defined as (Hochstadt, 1971)

$$J_\nu(w) = \left(\frac{w}{2}\right)^\nu \sum_{k=0}^{\infty} \frac{(-1)^k (w/2)^{2k}}{\Gamma(\nu + k + 1)k!}. \tag{2.103}$$

The *spherical Bessel* function, $j_l(w)$, of integer degree l is related to $J_\nu(w)$ by

$$j_l(w) = \sqrt{\frac{\pi}{2w}} J_{l+1/2}(w). \tag{2.104}$$

By using

$$(\alpha)_n = \frac{\Gamma(\alpha + n)}{\Gamma(\alpha)}, \tag{2.105}$$

in Eq 2.103, it is straightforward to show that $j_l(w)$ can be calculated as

$$j_l(w) = \frac{1}{2} \sum_{k=0}^{\infty} C_k^{(l)} \left(\frac{w}{2}\right)^{2k+l} \tag{2.106}$$

in which the coefficients, $C_k^{(l)}$, are given by

$$C_k^{(l)} = \frac{-2C_{k-1}^{(l)}}{k(2k + 2l + 1)}, \tag{2.107}$$

and where

$$C_0^{(l)} = \frac{1}{(1/2)_{l+1}}. \tag{2.108}$$

For $w \leq 2$, Eq 2.106 converges rapidly and the summation can be terminated once the desired level of accuracy has been obtained. For large w (i.e. up to around $w \simeq 100$), the spherical Bessel functions can be calculated using the recursion relation

$$j_l(w) = \frac{(2l - 1)}{w} j_{l-1}(w) - j_{l-2}(w) \tag{2.109}$$

with

$$j_0(w) = \frac{\sin w}{w} \tag{2.110}$$

and

$$j_1(w) = \frac{\sin w}{w^2} - \frac{\cos w}{w}. \tag{2.111}$$

Thus the spherical Bessel functions are seen to have a sinusoidal form which decays according to an inverse power of distance from the origin.

It can be shown that spherical Bessel functions are orthogonal in the sense that (Gottfried, 1966)

$$\int_0^\infty j_l(\beta r) j_l(\beta r')\, \beta^2 \mathrm{d}\beta = \frac{\pi}{2r}\delta(r - r'), \tag{2.112}$$

where

$$\delta(x) = \begin{cases} 1 \text{ if } x = 0, \\ 0 \text{ otherwise.} \end{cases} \tag{2.113}$$

If the spherical Bessel transform of a function $f(r)$ is defined as

$$\tilde{f}_l(\beta) = \sqrt{\frac{2}{\pi}} \int_0^\infty f(r)\, j_l(\beta r)\, r^2 \mathrm{d}r, \tag{2.114}$$

then using Eq 2.112 it can be shown that the inverse transform is given by

$$f(r) = \sqrt{\frac{2}{\pi}} \int_0^\infty \tilde{f}_l(\beta)\, j_l(\beta r)\, \beta^2 \mathrm{d}\beta. \tag{2.115}$$

Thus the spherical Bessel transform is seen to be its own inverse. Similarly, it is straightforward to prove that the spherical Bessel transforms of orthogonal functions are themselves orthogonal. In other words, if, for example,

$$\int_0^\infty R_{kl}(r) R_{nl}(r) r^2 \mathrm{d}r = \delta_{kn}, \tag{2.116}$$

then

$$\int_0^\infty \tilde{R}_{kl}(\beta) \tilde{R}_{nl}(\beta) \beta^2 \mathrm{d}\beta = \delta_{kn}. \tag{2.117}$$

These properties will be invoked in section 3.4 to develop analytic translation expressions for 3D SPF expansions.

2.2 3D Shape-Density Representations of Molecules

In quantum mechanical theories of matter, molecules are often treated as fixed arrangements of atomic nucleii surrounded by clouds of electrons. Mathematically, this may be represented as a superposition of electronic wave functions centred on the nuclear coordinates, which together define a probabilistic model of how the electrons are distributed throughout space. However, large protein molecules typically consist of hundreds of amino acids, thousands of atoms, and tens of thousands of electrons. Therefore, even with the largest supercomputers, it is essentially impossible to represent and calculate the properties of large proteins using *ab initio* electronic wave functions. Even small organic molecules often consist of tens of atoms and hundreds of electrons, and it becomes very

expensive to apply *ab initio* techniques to large numbers of small molecules. Hence approximate representations are often currently used to describe both small and large molecules.

A common and straightforward way to display the structures of molecules using computer graphics is simply to draw each atom of the molecule as a sphere of a given radius. The user then sees a space-filling union of all of the atomic spheres (see Figure 2.5). Usually, each type of atom is assigned an effective radius, the van der Waals (VDW) radius, calculated from crystal packing data. However, this "hard sphere" representation does not realistically model the fundamentally smooth nature of molecular electron density, and it does not provide an easy way to calculate accurately the total surface area or the volume of a molecule, for example.

However, Grant and Pickup (1995) showed that assigning a single Gaussian density function to each atomic position gives a remarkably effective way to describe the overall matter density of small molecules. For example, by writing

$$\rho_i(\underline{r}) = \alpha e^{-\beta(r/r_i)^2}, \tag{2.118}$$

where $\rho_i(\underline{r})$ is the density function for the i-th atom, r_i is its VDW radius, and where α and β are adjustable parameters, the overall matter density of a molecule of N_A atoms is then given by the sum of the atomic densities

$$\rho(\underline{r}) = \sum_{i=1}^{N_A} \rho_i(\underline{r}). \tag{2.119}$$

Furthermore, the overlap between pairs of such Gaussian functions has a particularly simple form (Boys, 1950). Grant and Pickup (1995) exploited this property to develop the very efficient ROCS (Rapid Overlay of Chemical Structures) small-molecule shape-matching program (Grant *et al.*, 1996). Following Grant and Pickup (1995), I use $\alpha = 2.70$ and $\beta = 2.3442$. Therefore, at a distance r_i from the centre of atom i, the density takes the constant value of $2.7e^{-2.3442} = 0.259$. Hence a good estimate of the VDW surface may be calculated by summing the atom density contributions at each node in a 3D grid, and by contouring the grid using a density threshold of $\rho = 0.259$. Figure 2.5 shows the contoured Gaussian density representation of lorazepam, a typical small drug molecule. Contouring is performed using my own implementation of the "marching tetrahedra" algorithm (Guéziec & Hummel, 1995).

In addition to calculating a smooth VDW surface, it is often also useful to be able to calculate the so-called solvent accessible surface (SAS). If a spherical probe molecule of a given radius, usually 1.4Å, is rolled over the VDW surface of a molecule without ever penetrating it, then the surface swept out by the probe's centre defines the SAS. Figure 2.6 shows a sketch of the SAS and the VDW surfaces.

When using the atomic Gaussian approach to represent molecular volumes, the SAS may be calculated by contouring a Gaussian surface in which the radius of each atom is extended by that of

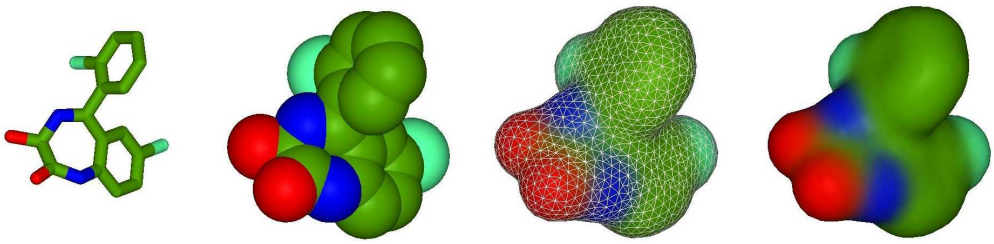

Figure 2.5: Illustrations of a small drug molecule, lorazepam, drawn using (from left to right) "licorice sticks" to represent the covalent bonds between the atoms, VDW spheres, a contoured Gaussian surface with surface triangles outlined in white, and the contoured Gaussian surface without outlining. In all representations, the atoms are coloured by atom type: carbon in green; oxygen in red; nitrogen in blue; chlorine in blue-green. In the Gaussian surfaces, surface triangles are coloured using a distance-weighted colour mixing rule which gives a smooth gradation of shades across the surface.

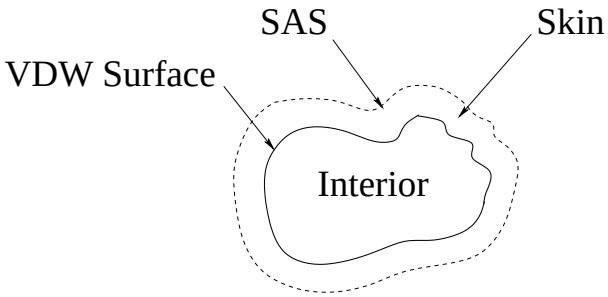

Figure 2.6: Schematic illustration of the molecular VDW surface, the SAS, and the *"surface skin"* and interior volumes.

the probe radius. Figure 2.7 shows the SAS for lorazepam. In Chapter 4 it is shown that the volume bounded by the SAS and VDW surfaces, which I call the "skin volume," plays an essential role in calculating protein shape complementarity during docking calculations.

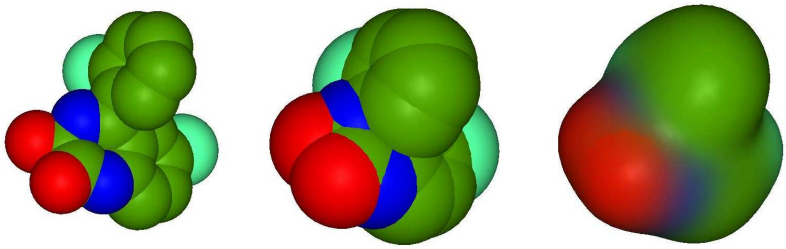

Figure 2.7: Comparison of (from left to right) VDW hard-sphere, extended VDW hard-sphere, and Gaussian SAS surfaces of lorazepam.

2.3 Icosahedral Tesselations of the Sphere

If one considers a conventional spherical grid, such as that formed by the lines of latitude and longitude on a map of the world, it is clear that the grid lines become very concentrated towards the north and south poles. A much fairer way to divide the surface of a sphere is to construct a spherical tessellation using the faces of a regular icosahedron, as illustrated in Figure 2.8. Here, icosahedral tessellations are calculated by constructing spherical triangles from the icosahedral faces and by using geodesic curves to subdivide the surface of each spherical triangle. This allows each icosahedral face to be divided into an integral number of subdivisions. I use the near-regular distribution of vertices of an icosahedral tessellation to sample rotational space evenly and to provide a fast way to estimate the integrals over the sphere which arise in the calculation of the SH molecular surface expansion coefficients, as described below.

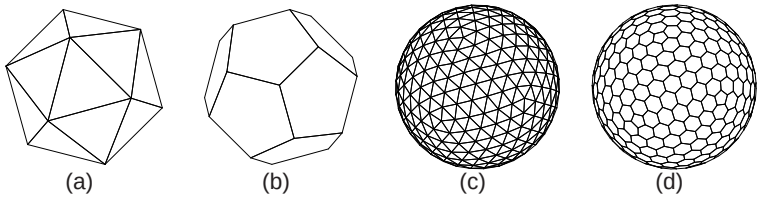

Figure 2.8: Illustrations of the icosahedron (a) and its dual, the dodecahedron (b). Subdividing the spherical triangular patches of the icosahedron gives the tessellation shown in (c). The dual tessellation (d) is obtained by connecting the centres of the triangular faces in (c). In this example, 6 subdivisions are made along each icosahedral edge to give an icosahedral tessellation of 362 vertices and 720 faces. The dual tessellation has 720 vertices which define 12 pentagonal and 350 hexagonal faces.

2.4 2D Spherical Harmonic Molecular Surfaces

The SHs may be used as orthogonal "building blocks" with which to construct functions parameterised by the spherical coordinates, (θ, ϕ). For example, the radial distance of a globular protein domain may be encoded as a sum of real SHs to order L using

$$r(\theta,\phi) = \sum_{l=0}^{L} \sum_{m=-l}^{l} a_{lm} y_{lm}(\theta,\phi), \tag{2.120}$$

where a_{lm} are the expansion coefficients. Multiplying each side of Eq 2.120 by $y_{kj}(\theta,\phi)$ and integrating over the sphere gives

$$\int_0^{2\pi} \int_0^{\pi} r(\theta,\phi) y_{kj}(\theta,\phi) \sin\theta d\theta d\phi = a_{lm}\delta_{kl}\delta_{jm}. \tag{2.121}$$

Thanks to the orthogonality of the basis functions, this reduces to

$$a_{lm} = \int_0^{2\pi} \int_0^{\pi} r(\theta, \phi) y_{lm}(\theta, \phi) \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{2.122}$$

In other words, each coefficient is uniquely determined by the degree of overlap with its corresponding basis function. Using the icosahedral tessellation of the sphere, this integral may be estimated as

$$a_{lm} = \sum_{i=1}^{N_V} r(\theta_i, \phi_i) y_{lm}(\theta_i, \phi_i) A_i, \tag{2.123}$$

where the sum is over the N_V vertices of the tessellation, (θ_i, ϕ_i) are the angular coordinates of the i-th tessellation vertex, and A_i is the area of the corresponding face in the dual mesh. Because finite area elements will not sum exactly to 4π, a somewhat more accurate way to calculate the coefficients is to use:

$$a_{lm} = \left(\frac{4\pi}{\sum_{i=1}^{N_V} A_i} \right) \sum_{i=1}^{N_V} r(\theta_i, \phi_i) y_{lm}(\theta_i, \phi_i) A_i. \tag{2.124}$$

Thus, armed with an icosahedral tessellation of the sphere, the task of calculating a SH surface largely reduces to the task of sampling the surface at each of the tessellation vertices and then using Eq 2.124 to calculate the SH expansion coefficients.

Figure 2.9 shows the VDW surface of lorazepam sampled onto an icosahedral tessellation of the sphere, along with the resulting smooth SH surface. Figure 2.10 shows the SH surfaces of lorazepam reconstructed at various expansion orders. Figure 2.11 shows the *L=16* SH surfaces of two protein molecules, an antibody and lysozyme, taken from the HyHel-5/lysozyme complex (PDB code 3HFL). Comparing these figures, it can be seen that low order SH expansions can capture the shapes of small globular molecules rather well, and using expansions above *L=8* gives very little apparent difference in resolution. On the other hand, the overall shapes of the larger protein molecules are clearly discernible with *L=16* expansions, but much of the atomic detail of these large molecules is missing. Furthermore, because SH representations must be single-valued, or star-like, with respect to radial rays from the chosen origin, the detailed shapes of cavities or pockets within a protein cannot be represented. Nonetheless, Chapter 4 shows that low order SH surface representations provide an effective way to search large databases of small molecules.

2.5 3D Spherical Polar Fourier Expansions

In order to capture the detailed shapes of large protein molecules sufficiently well to be able to perform docking calculations, it is necessary to augment the SH basis functions with suitable orthonormal radial basis functions. This essentially entails abandoning the notion of tangible surfaces, and adopting a mass-density model of protein shape. Hence for 3D representations, each scalar property of

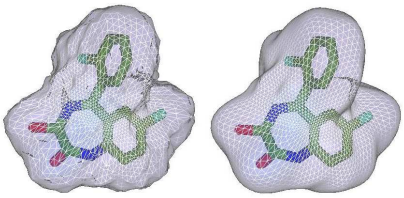

Figure 2.9: The VDW surface of lorazepam sampled onto an icosahedral tessellation (left) and the corresponding *L=16* SH surface (right) calculated using Eq 2.124 and reconstructed from the SH expansion coefficients. Note that the slightly "broken" appearance of the left-hand image is an artefact of the simple triangle depth-sorting algorithm that was used to achieve the transparency effect. The same depth-sorting algorithm is used on the right, but fewer incorrect surface triangle depths are calculated because the SH surface is smoother.

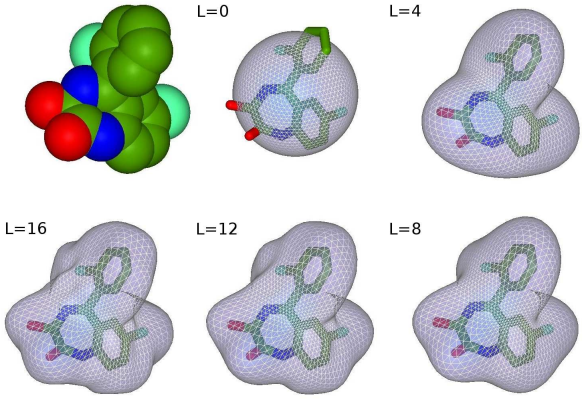

Figure 2.10: The SH surfaces of lorazepam at various expansion orders.

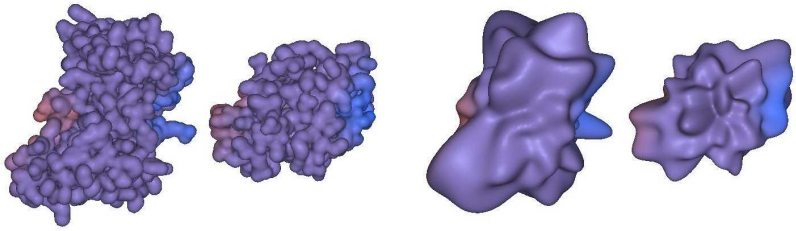

Figure 2.11: Comparison of a pair of 2D SH molecular surfaces to expansion order *L=16* (right) with the original atomic Gaussian density representation (left) of the HyHel-5 antibody Fv domain (large protein domain) and hen egg lysozyme (small domain). The two domains are separated by 15Å for clarity.

interest, $A(\underline{r})$, is encoded as a Fourier-like expansion of orthonormal SH and radial basis functions up to a given order N as

$$A(\underline{r}) = \sum_{n=1}^{N} \sum_{l=0}^{n-1} \sum_{m=-l}^{l} a_{nlm} R_{nl}(r) y_{lm}(\theta, \phi). \tag{2.125}$$

I use GTOs to represent steric shape and the more diffuse ETOs to represent electrostatic properties. The notation used here follows the quantum chemistry convention in which the radial index n, or principal quantum number, counts from unity. Hence the highest harmonic order and highest polynomial power in any individual coordinate is *L=N-1*.

It is worth noting that these are not the only type of radial function that could be used for 3D shape representation. For example, Mak *et al.* and Sael *et al.* recently described using Zernike polynomials (Novotni & Klein, 2003) with the SHs to construct rotationally invariant descriptors with which to compare protein shapes (Mak *et al.*, 2008; Sael *et al.*, 2008). However, rotationally invariant representations cannot be used to superpose or dock proteins because all orientational information is destroyed. The question of whether there might be better radial functions than the GL functions is considered in more detail in Section 5.9.

2.5.1 Calculating 3D Shape Density Functions

Following a similar approach to the 2D surface case, 3D shape-density representations of the molecular VDW interior (τ) and "surface skin" (σ) volumes may be defined as density functions:

$$\tau(\underline{r}) = \begin{cases} 1 & \text{if } \underline{r} \in \text{protein atom} \\ 0 & \text{otherwise,} \end{cases} \tag{2.126}$$

and

$$\sigma(\underline{r}) = \begin{cases} 1 & \text{if } \underline{r} \in \text{surface skin} \\ 0 & \text{otherwise.} \end{cases} \tag{2.127}$$

Writing these functions as SPF expansions to order N gives, for example,

$$\tau(\underline{r}) = \sum_{nlm} a^{\tau}_{nlm} R_{nl}(r)\, y_{lm}(\theta, \phi). \tag{2.128}$$

By sampling protein shapes onto a regular Cartesian grid, the expansion coefficients may be determined by summing non-zero cells in the grid:

$$\begin{aligned} a^{\tau}_{nlm} &= \int \tau(\underline{r}) R_{nl}(r)\, y_{lm}(\theta, \phi) \mathrm{d}V \\ &\simeq \sum_{k} R_{nl}(r_k)\, y_{lm}(\theta_k, \phi_k) \Delta V, \end{aligned} \tag{2.129}$$

where the summation runs over the non-zero grid cells, k, ΔV is the volume of each grid cell, and (r_k, θ_k, ϕ_k) are the polar coordinates of the centre of the k'th cell. I normally use a Cartesian sampling grid with 0.6Å^3 cells. Figure 2.12 shows some example SPF representations of the complex between the HyHel-5 antibody and hen egg lysozyme (PDB code 3HFL), calculated from the GTO expansion

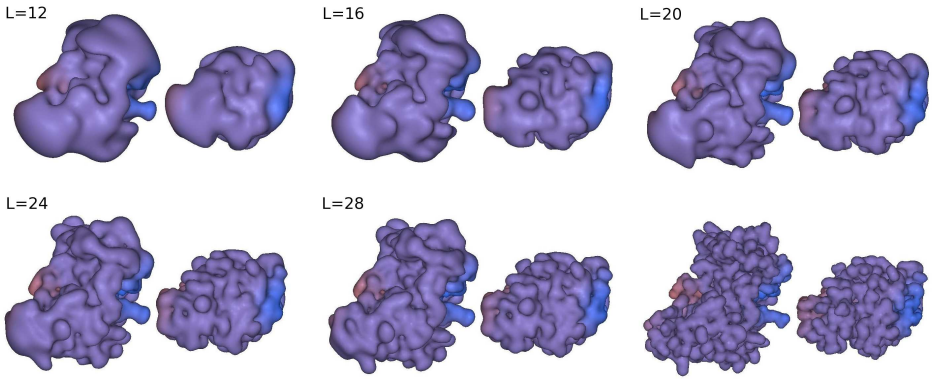

Figure 2.12: SPF steric density isosurfaces of various 3D GTO expansions for the complex between the HyHel-5 antibody Fv domain (left) and hen egg lysozyme (right). The subunits are separated by 15Å for clarity. The bottom right pair shows atomic Gaussian representations of the VDW surfaces from which the SPF expansions are derived.

coefficients at various orders *L=N-1*. This Figure shows that individual protein atoms begin to be resolved clearly with high order expansions of around *L=28*.

In a similar manner, the surface skin volume may be "voxelised" onto a Cartesian grid by using the SAS normal vectors calculated from the tetrahedral contouring algorithm to fill the volume between the SAS and the VDW surface with a large number of sampling spheres. Figure 2.13 illustrates this idea. The surface skin coefficients, a^{σ}_{nlm}, may then be computed by summing non-zero grid voxels as before.

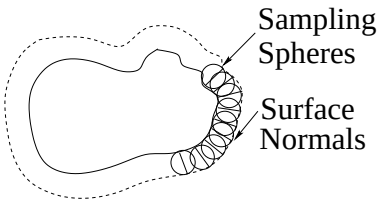

Figure 2.13: Schematic illustration of sampling the surface skin volume using small sampling spheres placed tangentially on the interior of the contoured and triangulated SAS.

2.5.2 Calculating Protein Electrostatic Properties

Classically, the electrostatic energy of a charge distribution, $\rho(\underline{r})$, under the influence of a potential, $\phi(\underline{r})$, is given by (Jackson, 1975)

$$E = \frac{1}{2}\int \rho(\underline{r})\phi(\underline{r})\mathrm{d}V. \tag{2.130}$$

In conventional molecular dynamics packages the parameters necessary to evaluate electrostatic interactions are often tabulated as point charges for each type of atom in a molecule. Hence it is convenient to use such point charges to calculate 3D SPF expansions. However, because these charge models usually assume the presence of hydrogen atoms, and because hydrogen atoms are normally not resolved in X-ray protein structures, it is first necessary to add polar hydrogens to a protein molecule. This is done automatically in the *Hex* program using standard amino acid geometries to infer or guess the hydrogen positions. Atom charges are then assigned from the AMBER parameter set (Weiner *et al.*, 1984). The SPF charge density coefficients may be calculated by equating an ETO expansion for $\rho(\underline{r})$ to the classical expression for the charge density due to a set of point charges, q_i, at positions $\underline{x}_i \equiv \underline{r}_i$ using

$$\rho(\underline{r}) = \sum_i q_i \, \delta(\underline{x} - \underline{x}_i) = \sum_{n'=1}^{N} \sum_{l'=0}^{n'-1} \sum_{m'=-l'}^{l'} a^{\rho}_{n'l'm'} S_{n'l'}(r) y_{l'm'}(\theta, \phi), \tag{2.131}$$

where $\delta(\underline{x})$ is the 3D Dirac delta

$$\delta(\underline{x}) = \begin{cases} 1 \text{ if } \underline{x} = (0,0,0), \\ 0 \text{ otherwise.} \end{cases} \tag{2.132}$$

Then, multiplying both sides of Eq 2.131 by $S_{nl}(r) y_{lm}(\theta, \phi)$ and integrating immediately gives the result

$$a^{\rho}_{nlm} = \sum_i q_i \, S_{nl}(r_i) y_{lm}(\theta_i, \phi_i). \tag{2.133}$$

The expansion coefficients for the *in vacuo* potential may be calculated from the charge density by solving Poisson's equation

$$\nabla^2 \phi(\underline{r}) = -4\pi \rho(\underline{r}). \tag{2.134}$$

Substituting the series expansion for each side, applying ∇^2 to the basis functions, multiplying both sides of the result by $S_{n'l'}(r) y_{l'm'}(\theta, \phi)$ and integrating gives

$$\sum_{n=l+1}^{N} a^{\phi}_{nlm} \int_0^{\infty} \big(S''_{nl}(r) + 2S'_{nl}(r)/r - l(l+1) S_{nl}(r)/r^2 \big) S_{n'l}(r) r^2 \mathrm{d}r = -4\pi a^{\rho}_{n'lm}, \tag{2.135}$$

where S' denotes $\partial S / \partial r$, etc. Then, integrating by parts the term in $S'_{nl}(r)$ gives

$$\sum_{n=l+1}^{N} a^{\phi}_{nlm} G^{(l)}_{nn'} = -4\pi a^{\rho}_{n'lm}, \tag{2.136}$$

where each element of $G^{(l)}$ has the symmetric form

$$G^{(l)}_{nn'} = -\int_0^{\infty} \big(S'_{nl}(r) S'_{n'l}(r) r^2 + l(l+1) S_{nl}(r) S_{n'l}(r) \big) \mathrm{d}r. \tag{2.137}$$

It can be seen that for each l and m, Eq 2.136 represents a set of simultaneous equations in the coefficients, $a^{\phi}_{n'lm}$, which can be determined by inverting each $G^{(l)}$ matrix. The elements of $G^{(l)}$ may be calculated by direct manipulation of the series expansion for the Laguerre polynomials. For example, writing

$$S_{nl}(r) = \sum_{k=0}^{n-l-1} D_{nlk}\, e^{-\rho/2} \rho^{l+k}, \tag{2.138}$$

where $\rho = 2\Lambda r$ and

$$D_{nlk} = (-1)^k \frac{\big((n-l-1)!(n+l+1)!\big)^{1/2}}{(n-l-k-1)!(k+2l+2)!k!}, \tag{2.139}$$

allows Eq 2.137 to be simplified term by term to give the symmetrical expression

$$G^{(l)}_{nn'} = \frac{1}{4} \sum_{k=0}^{n-l-1} \sum_{k'=0}^{n'-l-1} D_{nlk} D_{n'lk'} (2l+k+k')![(k-k')^2 - (k+k') - 2(2l+1)(l+1)]. \tag{2.140}$$

Each $G^{(l)}$ matrix may then be inverted using standard numerical techniques. With this approach, the charge density and potential expansions to order *N=30* can be calculated for a typical protein in under one second. It is worth noting that Poisson's equation is normally solved using numerical integration on a 3D grid, which can be expensive and error-prone. Provided the point charges are not too far away from the origin the analytic solution presented here is effectively exact up to the chosen expansion order. Figure 2.14 shows the ETO electrostatic potential on the surface of hen egg lysozyme protein (PDB code 1LZA) calculated to order *N=30*.

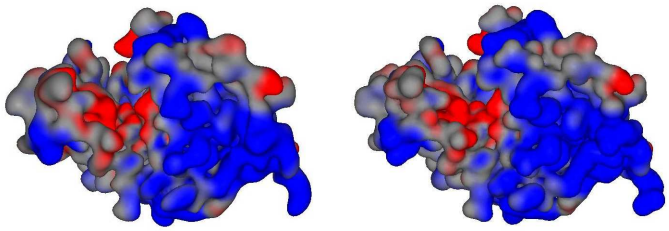

Figure 2.14: The ETO electrostatic potential calculated to order *N=30* for lysozome (PDB code 1LZA). The image on the left shows the electrostatic potential on the contoured *N=30* SPF density isosurface. The image on the right shows the same potential on the original contoured atomic Gaussian density. Red and blue colours represent negative and positive potentials, respectively. In these images the lysozyme is oriented to show its catalytic cleft (red) in the upper left region of the molecule.

Chapter 3

Spherical Polar Fourier Correlations

3.1 Operator Notation and 3D Coordinate Operations

In computer graphics applications, for example, in which complex graphical objects are often represented as lists of connected polygons, it is relatively straightforward to locate such objects within a scene by multiplying the coordinates of the component polygon vertices by suitably chosen 4×4 homogeneous rotation and translation matrices. On the other hand, in order to compare a pair of similar molecular shapes, or to dock a pair of complementary molecules, it is necessary to rotate and translate one or both molecules in order to find the relative orientation that gives the best superposition or contraposition, respectively. However, if the molecules are represented as Fourier-like expansions, and the goal is to transform such expansions directly, it is not so immediately obvious how the corresponding coordinate operations might be represented and implemented.

Therefore, it is useful to begin by defining abstract coordinate operators, $\hat{R}$ and $\hat{T}$, which will respectively rotate and translate Euclidean or Hilbert space objects or representations, as appropriate. For example, if $\hat{R}_z(\alpha)$ represents an operator which rotates a molecule (object) about the z axis by an amount α, and if the molecule is represented by an SPF expansion with coefficients a_{nlm}, then we wish to find suitably rotated coefficients a'_{nlm} such that the rotated molecule may be represented as

$$\hat{R}_z(\alpha)\sigma(\underline{r}) = \sum_{n=1}^{N}\sum_{l=0}^{n-1}\sum_{m=-l}^{l} a'_{nlm}R_{nl}(r)y_{lm}(\theta,\phi). \tag{3.1}$$

Now there is a direct analogy between rotations and translations in a Hilbert space and the corresponding coordinate operation in Euclidean space. For example, a positive “active” rotation of the expansion coefficients (object) is completely equivalent to an opposite “passive” rotation of the basis functions (coordinate axes). In other words, Eq 3.1 may equally be written as

$$\hat{R}_z(\alpha)\sigma(\underline{r}) \equiv \sigma(\hat{R}_z(\alpha)^{-1}\underline{r}) = \sum_{k=1}^{N}\sum_{j=0}^{k-1}\sum_{p=-j}^{j} a_{kjp}R_{kj}(r)y_{jp}(\theta,\phi-\alpha), \tag{3.2}$$

where the subscripts on the right-hand side have been re-labelled to facilitate the next step. If desired, the form of the rotation operator on the left-hand side may be instantiated as a 3×3 matrix

$$\hat{R}_z(\alpha)^{-1} = \underline{R}_z(-\alpha) = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.3}$$

Now, by equating the summations in Equations 3.1 and 3.2, multiplying both sides by $R_{nl}(r)y_{lm}(\theta,\phi)$ and integrating, one obtains

$$a'_{nlm} = a_{nlp} \int_0^{2\pi} \psi_m(\phi)\psi_p(\phi - \alpha). \tag{3.4}$$

In other words, by considering the *overlap* between the original and the transformed basis functions, one can calculate (at least in principle) the transformed expansion coefficients. In practice, this is straightforward for pure z-rotations, but it is much more difficult for general rotations and translations. Nonetheless, this is essentially the starting point used in Section 3.4 for calculating translations. However, it is first appropriate to consider rotations in further detail.

Because a rotation always involves an axis and a rotation angle, three angular parameters are necessary to describe a general rotation in 3D space (i.e. two angles fix the orientation of the axis, and the third angle specifies the amount of rotation about that axis). When working with SHs, it is usual to follow the Euler "z-y-z" convention for rotations in 3D space, in which an active rotation of an object is described by successive rotations of γ about the z axis, β about the y axis, and finally α about the z axis (again). This may be represented as

$$\hat{R}(\alpha,\beta,\gamma) = \hat{R}_z(\alpha)\hat{R}_y(\beta)\hat{R}_z(\gamma) \tag{3.5}$$

where the right-most operator is applied first. As will be seen below, rotations about the z axis are straightforward to implement for SH expansions. Hence the Euler z-y-z convention is a natural choice with which to represent 3D rotations. When applied to Cartesian coordinates, the general Euler rotation operator may be instantiated as

$$\underline{R}(\alpha,\beta,\gamma) = \begin{pmatrix} \cos\alpha\cos\beta\cos\gamma - \sin\alpha\sin\gamma & -\cos\alpha\cos\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta \\ \sin\alpha\cos\beta\cos\gamma + \cos\alpha\sin\gamma & \cos\alpha\cos\gamma - \sin\alpha\cos\beta\sin\gamma & \sin\alpha\sin\beta \\ -\sin\beta\cos\gamma & \sin\beta\sin\gamma & \cos\beta \end{pmatrix}. \tag{3.6}$$

Conversely, from the form of Eq 3.6, it can be seen that a general rotation matrix, $\underline{R}$, may be decomposed into the three Euler rotation angles according to

$$\begin{aligned} \beta &= \cos^{-1}(R_{22}) \\ \gamma &= \cos^{-1}(-R_{20}/\sin\beta) \\ \alpha &= \cos^{-1}(R_{02}/\sin\beta). \end{aligned} \tag{3.7}$$

The special cases of $R_{22} = 1$ and $R_{22} = -1$ may be resolved by putting $\gamma = 0$ and $\alpha = \cos^{-1}(R_{00})$ or $\alpha = \cos^{-1}(R_{11})$, respectively.

Although a given task may require the application of an arbitrary coordinate transformation to a polar Fourier expansion, a general translation is much harder to calculate than a general rotation in a SH basis. Hence, it is expedient to calculate translations only with respect to the z axis, and to define a general motion as a composition of one or more rotations and a single z-translation. For example, if $\underline{T}$ represents a general 4×4 homogeneous transformation,

$$\underline{T} = \begin{pmatrix} R_{00} & R_{01} & R_{02} & T_x \\ R_{10} & R_{11} & R_{12} & T_y \\ R_{20} & R_{21} & R_{22} & T_z \\ 0 & 0 & 0 & 1 \end{pmatrix}, \tag{3.8}$$

and if

$$\begin{aligned} r &= \sqrt{T_x^2 + T_y^2 + T_z^2} \\ \theta &= \cos^{-1}(T_z/r) \\ \phi &= \cos^{-1}(T_x/r\sin\theta) \end{aligned} \tag{3.9}$$

(where $\phi = 0$ if $\theta = 0$), then $\underline{T}$ may be decomposed as

$$\underline{T} = \underline{R}_2\, \underline{T}_z(r)\, \underline{R}_1, \tag{3.10}$$

where $\underline{T}_z(r)$ represents a pure translation of r along the positive z axis, and where $\underline{R}_1$ and $\underline{R}_2$ are pure rotations. It is then straightforward to see that

$$\underline{R}_2 = \underline{R}_z(\phi)\, \underline{R}_y(\theta) \tag{3.11}$$

and

$$\underline{R}_1 = \underline{T}_z(-r)\, \underline{R}_y(-\theta)\, \underline{R}_z(-\phi)\, \underline{T}. \tag{3.12}$$

Because $\underline{R}_1$ will normally resolve to three distinct Euler rotation angles, it can be seen that a general 3D transformation may be characterised by one translational and five angular parameters.

3.2 Addition Theorems and Correlations

An addition theorem is an algebraic relationship between the parameters of a function such as $f(a+b)$ in terms of the individual parameters, $f(a)$ and $f(b)$. A simple example of an addition theorem is

$$e^{i(\alpha+\beta)} = e^{i\alpha} e^{i\beta}. \tag{3.13}$$

In terms of operators, the action of applying two consecutive rotations of α and β about an axis in 3D space may also be considered as a kind of addition theorem. For example, considering Eq 3.13 it is natural to expect to be able to write

$$\hat{R}_z(\alpha+\beta) = \hat{R}_z(\alpha)\hat{R}_z(\beta), \tag{3.14}$$

and indeed using Euler's formula (Eq 2.14) it is straightforward to show that the corresponding 3×3 matrix representation of Eq 3.14 is given by

$$\begin{pmatrix} \cos(\alpha+\beta) & \sin(\alpha+\beta) & 0 \\ -\sin(\alpha+\beta) & \cos(\alpha+\beta) & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & \sin\beta & 0 \\ -\sin\beta & \cos\beta & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{3.15}$$

This matrix equation simultaneously embodies the "text book" geometric addition theorems

$$\cos(\alpha+\beta) = \cos\alpha\cos\beta - \sin\alpha\sin\beta \tag{3.16}$$

and

$$\sin(\alpha+\beta) = \sin\alpha\cos\beta + \cos\alpha\sin\beta. \tag{3.17}$$

More sophisticated addition theorems have been found which involve the special functions. For example, a translational addition theorem for the regular solid SHs may be expressed as (Biedenharn & Louck, 1981):

$$Y_{lm}(\underline{x}+\underline{T}) = \sum_{kj}\left[\frac{4\pi(2l+1)}{(2l-2k+1)(2k+1)}\binom{l+m}{k+j}\binom{l-m}{k-j}\right]^{1/2} Y_{kj}(\underline{x})Y_{l-k,m-j}(\underline{T}), \tag{3.18}$$

where the summation runs over all values of k and j for which the factorials are well defined.

More generally, two coordinate systems $\underline{r} = (r,\theta,\phi)$ and $\underline{r}' = \underline{r} - \underline{T} = (r',\theta',\phi)$, as illustrated in Figure 3.1, may be related functionally using Raleigh's plane wave addition theorem (Bransden & Joachain, 1997). For example, multiplying the vector equation

$$\underline{r} = \underline{T} + \underline{r}' \tag{3.19}$$

by an arbitrary complex vector $i\underline{k}$ and exponentiating each side gives:

$$e^{i\underline{k}.\underline{r}} = e^{i\underline{k}.\underline{T}}\, e.^{i\underline{k}.\underline{r}'} \tag{3.20}$$

Writing the direction vectors in spherical polar coordinates, $\underline{k} = (\beta,\Theta,\Phi)$, $\underline{r} = (r,\theta,\phi)$, $\underline{r}' = (r',\theta',\phi')$, and $\underline{T} = (R,\gamma,\delta)$, Raleigh's addition theorem relates the two coordinate systems according to:

$$e^{i\underline{k}.\underline{r}} = 4\pi\sum_{l=0}^{\infty}\sum_{m=-l}^{l} i^l j_l(\beta r)Y_{lm}(\Theta,\Phi)^*Y_{lm}(\theta,\phi). \tag{3.21}$$

When the translation is restricted to the positive z direction it can be shown that Raleigh's equation may be simplified to obtain a spherical Bessel addition theorem:

$$j_l(\beta r)Y_{lm}(\theta,\phi) = \sum_{l'=0}^{\infty}\sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} j_k(\beta R) j_{l'}(\beta r')Y_{l'm}(\theta',\phi), \tag{3.22}$$

where the angular coefficient $A_k^{(ll'|m|)}$ is given by

$$A_k^{(ll'|m|)} = (-1)^{(k+l'-l)/2+m}(2k+1)\big[(2l+1)(2l'+1)\big]^{1/2}\begin{pmatrix} l & l' & k \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} l & l' & k \\ m & \overline{m} & 0 \end{pmatrix}, \tag{3.23}$$

From the permutational symmetries of the second 3–j symbol, the right hand side is independent of the sign of m thus justifying the use of $|m|$ to label the matrix elements. Because the first 3–j symbol vanishes whenever $l + l' + k$ is odd, it can be seen that the non-vanishing coefficients are always real and that the summation on k need only be calculated for even increments with $k = |l - l'|, |l - l'| + 2, ..., l + l'$. For a negative translation, it can be shown that an additional factor of $(-1)^{l-l'}$ appears in the above expression. For full details, see e.g. Appendix A of Ritchie (2005).

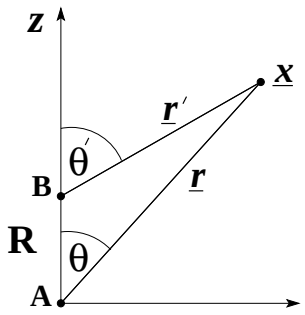

Figure 3.1: Illustration of the coordinate systems used to represent translations. The position of a point $\underline{x}$ with respect to the coordinate system whose origin is at position A is $\underline{x} = \underline{r} = (r, \theta, \phi)$, whereas the same point has the coordinates $\underline{x} = \underline{r}' = (r', \theta', \phi)$ with respect to the coordinate system whose origin is at position B. The two coordinate systems A, and B, are related by a translation R in the z direction.

Addition theorems are often very useful when calculating correlations. In classical Fourier analysis, the *correlation* between a pair of functions, $f(\phi)$ and $g(\phi)$, is conventionally defined as

$$(f * g)(\alpha) \equiv \int_0^{2\pi} f(\phi)^* g(\phi + \alpha)\mathrm{d}\phi. \tag{3.24}$$

In other words, a correlation is the degree of overlap between one function and a shifted version of the other function. Of course, for molecular shape matching and docking purposes, the aim is to calculate overlap integrals in 3D space in which one or both of the functions have been rotated or translated by specific amounts. However, the concise notation used in Eq 3.24 for one-dimensional correlations

cannot easily be extended to describe multi-dimensional correlations unambiguously. Hence I prefer to use the operator notation defined above. Thus, for example, Eq 3.24 would be expressed as

$$(f * g)(\alpha) \equiv \int_0^{2\pi} f(\phi)^* [g(\hat{R}(\alpha)\phi)] \mathrm{d}\phi \equiv \int_0^{2\pi} f(\phi)^* [\hat{R}(-\alpha) g(\phi)] \mathrm{d}\phi. \tag{3.25}$$

3.3 Rotating Spherical Polar Fourier Expansions

3.3.1 The Wigner Rotation Matrices

It can be shown that the SH functions of each order, l, transform amongst themselves under a general Euler rotation according to (Rose, 1957; Biedenharn & Louck, 1981)

$$\hat{R}(\alpha,\beta,\gamma) Y_{lm}(\theta,\phi) = \sum_{m'} Y_{lm'}(\theta,\phi) D^{(l)}_{m'm}(\alpha,\beta,\gamma) \tag{3.26}$$

where $\hat{R}(\alpha,\beta,\gamma)$ represents a rotation operator expressed in terms of the Euler angle parameterisation (α,β,γ). The unitary rotation matrices $D^{(l)}(\alpha,\beta,\gamma)$ are originally due to Wigner (1939). However, it seems he never published a full derivation. Probably the most elegant and direct way to calculate general rotations of SHs is to use a Boson operator technique (Sakurai, 1994). The result is (Biedenharn & Louck, 1981)

$$D^{(l)}_{m'm}(\alpha,\beta,\gamma) = e^{-im'\alpha} d^{(l)}_{m'm}(\beta) e^{-im\gamma}, \tag{3.27}$$

where

$$\begin{aligned} d^{(l)}_{m'm}(\beta) = {} & \big[(l+m')!(l-m')!(l+m)!(l-m)!\big]^{1/2} \times \\ & \sum_{k=\max(0,m-m')}^{\min(l-m',l+m)} \frac{(-1)^{k+m'-m} (\cos\beta/2)^{2l+m-m'-2k} (\sin\beta/2)^{2k+m'-m}}{(l+m-k)!k!(m'-m+k)!(l-m'-k)!}. \end{aligned} \tag{3.28}$$

The $d^{(l)}$ matrix elements have the useful symmetries

$$\begin{aligned} d^{(l)}_{m'm}(\beta) &= (-1)^{m'-m} d^{(l)}_{mm'}(\beta) \\ &= (-1)^{m'-m} d^{(l)}_{\overline{m}'\overline{m}}(\beta). \end{aligned} \tag{3.29}$$

From a programming point of view, this means that approximately three quarters of the elements of each $d^{(l)}$ matrix may be calculated trivially. When $|m'| = l$ or $|m| = l$, the summation in Eq 3.28 reduces to a single term and one obtains, for example,

$$d^{(l)}_{m'l} = \left[\frac{(2l)!}{(l-m')!(l+m')!}\right]^{1/2} (\cos\beta/2)^{l+m'} (\sin\beta/2)^{l-m'}. \tag{3.30}$$

Like the other special functions, there exist several three-term recursion relations which may be exploited for efficient calculations. For example, except at the poles, the relation

$$\frac{2(m\cos\beta - m')}{\sin\beta} d^{(l)}_{m'm} = \big[(l+m+1)(l-m)\big]^{1/2} d^{(l)}_{m',m+1} + \big[(l-m+1)(l+m)\big]^{1/2} d^{(l)}_{m',m'-1} \tag{3.31}$$

may be used to calculate consecutive matrix elements efficiently starting from Eq 3.30.

3.3.2 Real Wigner Rotation Matrices

Linear combinations of complex SHs such as Eq 2.82 preserve rotational symmetry, and the real SH functions also transform amongst themselves under rotation. This behaviour may be represented as

$$\hat{R}(\alpha,\beta,\gamma)y_{lm}(\theta,\phi) = \sum_{m'} y_{lm'}(\theta,\phi)R^{(l)}_{m'm}(\alpha,\beta,\gamma). \tag{3.32}$$

where $R^{(l)}$ is a real rotation matrix. If Eq. 2.82 is written as a sum,

$$y_{lm}(\theta,\phi) = \sum_{m'} U^{(l)}_{mm'} Y_{lm'}(\theta,\phi), \tag{3.33}$$

then it can be shown that for each order, l, the real rotation matrix, $\underline{R}$, is given by the matrix equation

$$\underline{R}^{(l)} = \underline{U}^{(l)}\underline{D}^{(l)}\underline{U}^{(l)\dagger}, \tag{3.34}$$

where $\underline{U}^{\dagger}$ is the complex conjugate transpose of $\underline{U}$. Recognising that all non-diagonal elements of $\underline{U}$ are zero, it is relatively straightforward, but tedious, to simplify Eq. 3.34 symbolically. The result is

$$R^{(l)}_{m'm} = \begin{cases} d^{(l)}_{m'm}(\beta)\cos(m\gamma + m'\alpha) + (-1)^{m'} d^{(l)}_{\overline{m}'m}(\beta)\cos(m\gamma - m'\alpha) & ;\ m'>0, m>0 \\ d^{(l)}_{0m}(\beta)\sqrt{2}\cos(m\gamma) & ;\ m'=0, m>0 \\ (-1)^{m'+1} d^{(l)}_{m'm}(\beta)\sin(m\gamma + m'\alpha) + d^{(l)}_{\overline{m}'m}(\beta)\sin(m\gamma - m'\alpha) & ;\ m'<0, m>0 \\ d^{(l)}_{m'0}(\beta)\sqrt{2}\cos(m'\alpha) & ;\ m'>0, m=0 \\ d^{(l)}_{00}(\beta) & ;\ m'=0, m=0 \\ (-1)^{m'+1} d^{(l)}_{m'0}(\beta)\sqrt{2}\sin(m'\alpha) & ;\ m'<0, m=0 \\ (-1)^{m} d^{(l)}_{m'm}(\beta)\sin(m\gamma + m'\alpha) + (-1)^{m+m'} d^{(l)}_{\overline{m}'m}(\beta)\sin(m\gamma - m'\alpha) & ;\ m'>0, m<0 \\ (-1)^{m} d^{(l)}_{0m}(\beta)\sqrt{2}\sin(m\gamma) & ;\ m'=0, m<0 \\ (-1)^{m+m'} d^{(l)}_{m'm}(\beta)\cos(m\gamma + m'\alpha) + (-1)^{m+1} d^{(l)}_{\overline{m}'m}(\beta)\cos(m\gamma - m'\alpha) & ;\ m'<0, m<0. \end{cases} \tag{3.35}$$

Setting $\beta = \gamma = 0$ gives a single rotation, α, about the z-axis which has a particularly simple form. Since

$$\hat{R}(\alpha,0,0)\ y_{lm}(\theta,\phi) = y_{lm}(\theta,\phi-\alpha), \tag{3.36}$$

the identities

$$\cos(\alpha \pm \beta) = \cos\alpha\cos\beta \mp \sin\alpha\sin\beta \tag{3.37}$$

and

$$\sin(\alpha \pm \beta) = \sin\alpha\cos\beta \pm \cos\alpha\sin\beta \tag{3.38}$$

may be used to show that

$$\hat{R}(\alpha, 0, 0)\ y_{lm}(\theta, \phi) = y_{lm}(\theta, \phi) \cos m\alpha + y_{l\overline{m}}(\theta, \phi) \sin m\alpha. \tag{3.39}$$

With a little more work, the identity

$$d^{(l)}_{m'm}(\beta = 0) = \delta_{m'm} \tag{3.40}$$

can be substituted into the rotation matrix element expressions (Eq 3.35), and Eq 3.32 may be simplified to obtain the same result. For a general rotation, in which $\beta \neq 0$, all rotation matrix elements must be calculated but the amount of computation can be reduced greatly by using the symmetries of the $d^{(l)}$ matrices (Eq 3.29).

3.4 Translating Spherical Polar Fourier Expansions

Much of QM is concerned with calculating integrals over products of electronic wave functions centred on different nuclear coordinates. However, although these integrals depend on the distance(s) between the nuclear centres, there is very little literature on the notion of relating such integrals to the action of a translation operator. Instead, it seems that the usual practice in QM is to re-calculate any necessary overlap integrals whenever any of the nuclear coordinates change. No doubt, this is the correct way to proceed in QM where the task is to evaluate as accurately as possible very large numbers of relatively low order multi-centre integrals. However, for protein docking and molecular shape-matching purposes, we need to calculate the overlap between considerably higher order basis functions about just two expansion centres, and we need to repeat these calculations over potentially very many distinct rotational orientations of one or both molecules. Therefore, it is imperative to develop an explicit matrix representation of the translation operator in order to avoid needlessly having to re-calculate very many high order overlap integrals during a correlation search.

3.4.1 Overlap Integrals as Translation Matrix Elements

A straightforward way to find the general form of the translation matrices is firstly to consider the overlap between a fixed "body" A, or scalar function $A(\underline{r})$, and a moving body B, or function $B(\underline{r})$, under an active translation of B by $\underline{T} = (R, 0, 0)$ along the positive z axis, as illustrated above in Figure 3.1. Symbolically, this may be expressed as

$$C(\underline{T}) = \int A(\underline{r}) B(\underline{T}^{-1}\underline{r})\, \mathrm{d}V. \tag{3.41}$$

Substituting the expansions of $A(\underline{r})$ and $B(\underline{r})$ gives

$$C(\underline{T}) = \sum_{nlm} \sum_{n'l'm'} a_{nlm} b_{n'l'm'} \int R_{nl}(r) y_{lm}(\theta, \phi) R_{n'l'}(r') y_{l'm'}(\theta', \phi')\, \mathrm{d}V \tag{3.42}$$

where $\underline{r} = (r, \theta, \phi)$ and $\underline{r}' = \underline{r} - \underline{T} = (r', \theta', \phi')$, and where the shorthand notation $\sum_{nlm}$ etc. is used to indicate summation over the subscript ranges given in Eq 2.125. In this case, ϕ and ϕ' remain coincident so the circular functions, $\varphi_m(\phi)$, may be integrated out, and Eq 3.42 reduces to a sum over 2D integrals in the (r, θ) plane. Because the value of each of these integrals depends only on the distance R and is independent of the sign of m (see below), we may write:

$$T^{(|m|)}_{nl,n'l'}(R) = \int R_{nl}(r)\vartheta_{lm}(\theta)R_{n'l'}(r')\vartheta_{l'm}(\theta')r^2 \mathrm{d}r \sin\theta \mathrm{d}\theta, \tag{3.43}$$

and

$$C(R) = \sum_{nlm} \sum_{n'l'm'} a_{nlm} b_{n'l'm'} T^{(|m|)}_{nl,n'l'}(R)\delta_{mm'}, \tag{3.44}$$

and interpret each $T^{(|m|)}_{nl,n'l'}(R)$ as a matrix element of the translation operator. For example, from Eq 3.44 it can be seen that the two sums

$$b^R_{nlm} = \sum_{n'l'} T^{(|m|)}_{nl,n'l'}(R) b_{n'l'm} \tag{3.45}$$

and, after re-labeling the subscripts,

$$a^R_{nlm} = \sum_{n'l'} T^{(|m|)}_{n'l',nl}(R) a_{n'l'm} \tag{3.46}$$

represent a positive translation of the body B, or equivalently a negative translation of the body A, respectively. The translation matrices are obviously five-dimensional quantities. However, because they do not depend on the sign of m, it is useful to consider each matrix as being composed of $\sum_{m=0}^{N-1} = N$ two-dimensional arrays, each indexed by $\sum_{nl} = N(N+1)/2$ possible values for each pair of nl subscripts. The matrix elements vanish trivially where $|m| > l$. Furthermore, the notation used here is intended to be consistent with the usual convention for the complex and real SH rotation matrix elements, $D^{(l)}_{m'm}(\alpha, \beta, \gamma)$ and $R^{(l)}_{m'm}(\alpha, \beta, \gamma)$, respectively, in the sense that a positive z-translation of the basis functions is expressed as

$$R_{nl}(r')y_{lm}(\theta', \phi) = \sum_{n'=1}^{\infty} \sum_{l'=0}^{n'-1} T^{(|m|)}_{n'l',nl}(R) R_{n'l'}(r) y_{l'm}(\theta, \phi). \tag{3.47}$$

In the rest of this section it will be shown that the translation matrix elements for SPF basis functions may be calculated as sums over 1D inverse spherical Bessel transforms. Firstly, from Section 2.1.14, if the spherical Bessel transform of $R_{nl}(r)$ is defined as

$$\tilde{R}_{nl}(\beta) = (2/\pi)^{1/2} \int_0^{\infty} R_{nl}(r) j_l(\beta r) r^2 \mathrm{d}r, \tag{3.48}$$

then the inverse transform is given by

$$R_{nl}(r) = (2/\pi)^{1/2} \int_0^{\infty} \tilde{R}_{nl}(\beta) j_l(\beta r) \beta^2 \mathrm{d}\beta. \tag{3.49}$$

Next, multiplying both sides of the spherical Bessel addition theorem, Eq 3.22, by $(2/\pi)^{1/2}\tilde{R}_{nl}(\beta)\beta^2$ and integrating with respect to β (i.e. applying the above inverse transform) gives

$$R_{nl}(r)Y_{lm}(\theta,\phi) = \left(\frac{2}{\pi}\right)^{1/2} \sum_{l'=0}^{\infty} \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} \int_0^{\infty} \tilde{R}_{nl}(\beta) j_k(\beta R) j_{l'}(\beta r') \beta^2 \mathrm{d}\beta \; Y_{l'm}(\theta',\phi). \tag{3.50}$$

Then, multiplying each side by $R_{n'j'}(r')Y_{j'm'}(\theta',\phi)$ and integrating over all space in the corresponding variables gives

$$T^{(|m|)}_{n'j',nl}(R) = \left(\frac{2}{\pi}\right)^{1/2} \sum_{l'=0}^{\infty} \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} \int_0^{\infty} \int_0^{\infty} \delta_{j'l'} R_{n'j'}(r') j_{l'}(\beta r') \, \tilde{R}_{nl}(\beta) j_k(\beta R) \beta^2 \mathrm{d}\beta \, r'^2 \mathrm{d}r'. \tag{3.51}$$

Finally, recognising the integral in r' as the spherical Bessel transform of $R_{n'l'}(r')$, the result reduces to a finite sum of terms:

$$T^{(|m|)}_{n'l',nl}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} \int_0^{\infty} \tilde{R}_{n'l'}(\beta) \tilde{R}_{nl}(\beta) j_k(\beta R) \beta^2 \mathrm{d}\beta. \tag{3.52}$$

This generalises the expression given by Danos and Maximon (1965) for translating multipole expansions to the more general case for arbitrary orthonormal radial functions, $R_{nl}(r)$. By similar arguments, it can also be shown that

$$T^{(|m|)}_{nl,n'l'}(R) = T^{(|m|)}_{n'l',nl}(-R) = (-1)^{l'-l} T^{(|m|)}_{n'l',nl}(R). \tag{3.53}$$

Consequently, nearly half of all matrix elements can be found by symmetry. Given that the original basis functions form a complete orthonormal set, it is straightforward to show that the translation matrices are also orthonormal in sense that

$$\sum_{n'=1}^{\infty} \sum_{l'=0}^{n'-1} T^{(|m|)}_{n'l',nl}(R) T^{(|m|)}_{n'l',n''l''}(R) = \delta_{nn''}\delta_{ll''}. \tag{3.54}$$

Evaluating this equation provides a convenient way to verify the correctness of the following calculations.

The following sections will develop explicit closed form expressions for the translation matrix elements for the GTO and ETO basis functions.

3.4.2 The GTO Translation Matrix Elements

From Eq 2.98, the normalised GTO radial functions are given by

$$R_{nl}(r) = \left[\frac{2}{\lambda^{3/2}\pi^{1/2}} \frac{(n-l-1)!}{(1/2)_n}\right]^{1/2} e^{-\rho^2/2} \rho^l L_{n-l-1}^{(l+1/2)}(\rho^2) \; ; \quad \rho^2 = r^2/\lambda. \tag{3.55}$$

The GTO functions are eigenfunctions of the spherical Bessel transform (derived from (Erdélyi *et al.*, 1953c), p42, Eq 3)

$$\tilde{R}_{nl}(\beta) = (-1)^{n-l-1}\left[\frac{2\lambda^{3/2}}{\pi^{1/2}}\frac{(n-l-1)!}{(1/2)_n}\right]^{1/2} e^{-x^2/2}x^l L_{n-l-1}^{(l+1/2)}(x^2), \tag{3.56}$$

where $x^2 = \lambda\beta^2$. Here, it is convenient to use Eq 2.89 to expand Eq 3.56 as a power series

$$\tilde{R}_{nl}(\beta) = \left[\frac{4\lambda^{3/2}}{\pi^{1/2}}\right]^{1/2} \sum_j X_{nlj} e^{-x^2/2} x^{2j+l}, \tag{3.57}$$

where $\sum_j$ serves as shorthand notation for $\sum_{j=0}^{n-l-1}$, and where the coefficients X_{nlj} are given by

$$X_{nlj} = \left[\frac{(n-l-1)!(1/2)_n}{2}\right]^{1/2} \frac{(-1)^{n-l-j-1}}{j!(n-l-j-1)!(1/2)_{l+j+1}}. \tag{3.58}$$

Substituting Eq 3.57 twice into Eq 3.52 and collecting coefficients of x^{2k} using

$$C_k^{(nl,n'l')} = \sum_j \sum_{j'} \delta_{k,j+j'} X_{nlj} X_{n'l'j'} \tag{3.59}$$

gives for the GTO translation matrix elements

$$T_{n'l',nl}^{(|m|)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} \sum_{j=0}^{n-l+n'-l'-2} C_j^{(nl,n'l')} \frac{4}{\pi^{1/2}} \int_0^\infty x^{2j+l+l'} j_k(xR/\lambda^{1/2}) x^2 \mathrm{d}x. \tag{3.60}$$

Applying the relation (from (Erdélyi *et al.*, 1953c), p30, Eq 13)

$$\frac{4}{\pi^{1/2}} \int_0^\infty e^{-x^2} x^{2m+k} j_k(xy) x^2 \mathrm{d}x = m! e^{-y^2/4} (y^2/4)^{k/2} L_m^{(k+1/2)}(y^2/4), \tag{3.61}$$

then gives the final analytic result

$$T_{n'l',nl}^{(|m|)}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} \sum_{j=0}^{n-l+n'-l'-2} C_j^{(nl,n'l')} M! e^{-R^2/4\lambda} (R^2/4\lambda)^{k/2} L_M^{(k+1/2)}(R^2/4\lambda), \tag{3.62}$$

where $M = j + (l + l' - k)/2$.

3.4.3 The ETO Translation Matrix Elements

From Eq 2.101, the normalised ETO radial functions are given by

$$S_{nl}(r) = \left[(2\Lambda)^3 \frac{(n-l-1)!}{(n+l+1)!}\right]^{1/2} e^{-\rho/2} \rho^l L_{n-l-1}^{(2l+2)}(\rho), \tag{3.63}$$

where $\rho = 2\Lambda r$ with scale factor Λ. I set $\Lambda = 1/2$ for protein-protein electrostatic calculations (Ritchie & Kemp, 2000). Using an argument based on orthogonality, Keister and Polyzou (1997) recently proved that the spherical Bessel transform of these functions may be written in terms of the Jacobi polynomials, $P_k^{(\alpha,\beta)}(t)$

$$\tilde{S}_{nl}(\beta) = \frac{2}{(1/2)_n}\left[\frac{(n-l-1)!(n+l+1)!}{\pi\Lambda^3}\right]^{1/2}\frac{s^l}{(s^2+1)^{l+2}}P_{n-l-1}^{(l+3/2,l+1/2)}\left(\frac{s^2-1}{s^2+1}\right), \tag{3.64}$$

where $s = \beta/\Lambda$. Following a similar treatment to the GTO case, the shifted series expansion (Keister & Polyzou, 1997)

$$P_k^{(\mu,\nu)}(t) = \frac{1}{k!}\frac{\Gamma(k+\mu+1)}{\Gamma(k+\mu+\nu+1)}\sum_{j=0}^{k}(-1)^j\binom{k}{j}\frac{\Gamma(k+j+\mu+\nu+1)}{\Gamma(j+\mu+1)}\left(\frac{1-t}{2}\right)^j \tag{3.65}$$

may be used to collect factors of $1/(s^2+1) = (1-t)/2$ to write Eq 3.64 as a power series

$$\tilde{S}_{nl}(\beta) = \left[\frac{2}{\pi\Lambda^3}\right]^{1/2}\sum_j Y_{nlj}\frac{s^l}{(s^2+1)^{l+j+2}} \tag{3.66}$$

where

$$Y_{nlj} = \left[\frac{1}{2}\frac{(n-l-1)!}{(n+l+1)!}\right]^{1/2}\frac{(-1)^j(2n+1)(n+l+j+1)!}{j!(n-l-j-1)!(1/2)_{l+j+2}}. \tag{3.67}$$

Substituting Eq 3.66 twice into Eq 3.52 and collecting coefficients of $1/(s^2+1)^k$ using

$$D_k^{(nl,n'l')} = \sum_j\sum_{j'}\delta_{k,j+j'}Y_{nlj}Y_{n'l'j'} \tag{3.68}$$

gives for the ETO translation matrix elements

$$U_{n'l',nl}^{(|m|)}(R) = \sum_{k=|l-l'|}^{l+l'}A_k^{(ll'|m|)}\sum_{j=0}^{n-l+n'-l'-2}D_j^{(nl,n'l')}\frac{2}{\pi}\int_0^\infty\frac{s^{2M+k}}{(s^2+1)^{J+2}}j_k(s\Lambda R)s^2\mathrm{d}s, \tag{3.69}$$

where $M = (l+l'-k)/2$ and $J = j+l+l'+2$. It is shown in Appendix B of Ritchie (2005) that the remaining integral may be calculated in closed form as

$$\frac{2}{\pi}\int_0^\infty\frac{s^{2M+k}}{(s^2+1)^{J+2}}j_k(s\Lambda R)s^2ds = \sum_{q=0}^{M}\binom{M}{q}\frac{(-1)^{M+q}}{2^{J+1-q}(J+1-q)!}(\Lambda R)^k\hat{k}_{J-k-q+1/2}(\Lambda R), \tag{3.70}$$

where $\hat{k}_\sigma(z)$ is a reduced Bessel function of the second kind. For half-integral degree, these functions may be calculated using the recurrence relations (Weniger & Steinborn, 1983)

$$\begin{aligned}\hat{k}_{1/2}(z) &= e^{-z},\\ \hat{k}_{3/2}(z) &= (1+z)e^{-z},\\ \hat{k}_{n+3/2}(z) &= (2n+1)\hat{k}_{n+1/2}(z) + z^2\hat{k}_{n-1/2}(z).\end{aligned} \tag{3.71}$$

Thus, the ETO translation matrix elements may also be calculated analytically, although compared to the GTO basis an additional inner summation is necessary.

3.4.4 Non-Orthogonal Translation Matrices

Translations of SPF expansions in both the GTO and ETO bases can be computed more economically by eliminating the inner summation on the subscript j in Eq.s 3.62 and 3.69. This is equivalent to calculating overlap integrals that correspond to expansions of non-orthogonal radial basis functions. For example, substituting Eq 3.57 into Eq 3.52 and applying Eq 3.61 directly gives the factorisation

$$T^{(|m|)}_{n'l',nl}(R) = \sum_{j'}\sum_{j} X_{n'l'j'}\bar{T}^{(|m|)}_{j'l',jl}(R)X_{nlj}, \tag{3.72}$$

where each $\bar{T}^{(|m|)}_{j'l',jl}(R)$ is an overlap integral in a non-orthogonal basis

$$\bar{T}^{(|m|)}_{j'l',jl}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} M! e^{-R^2/4\lambda}(R^2/4\lambda)^{k/2} L_M^{(k+1/2)}(R^2/4\lambda) \tag{3.73}$$

now with $M = j+j'+(l+l'-k)/2$. This corresponds to expanding $R_{nl}(r)$ as a sum of non-orthogonal functions, $\bar{R}_{nl}(r)$

$$R_{nl}(r) = (-1)^{n-l-1}\sum_{j} X_{nlj}\bar{R}_{jl}(r). \tag{3.74}$$

It can be shown that these functions correspond to the non-orthogonal GL basis proposed by Chiu and Moharerrzadeh (1999). With this factorisation, translated expansion coefficients, a^R_{nlm}, in the original orthogonal basis may be calculated using the sequence

$$\bar{a}_{jlm} = \sum_{n} X_{nlj} a_{nlm}, \tag{3.75}$$

$$\bar{a}^R_{jlm} = \sum_{j'l'} \bar{T}^{(|m|)}_{jl,j'l'}(R)\bar{a}_{j'l'm}, \tag{3.76}$$

$$a^R_{nlm} = \sum_{j} X_{nlj}\bar{a}^R_{jlm}. \tag{3.77}$$

In a similar manner, ETO translations may be calculated in a non-orthogonal basis by substituting Eq 3.66 into Eq 3.52 and collecting powers of $1/(s^2+1)$ directly to give

$$U^{(|m|)}_{n'l',nl}(R) = \sum_{j'}\sum_{j} Y_{n'l'j'}\bar{U}^{(|m|)}_{j'l',jl}(R)Y_{nlj} \tag{3.78}$$

where the non-orthogonal matrix elements, $\bar{U}^{(|m|)}_{j'l',jl}(R)$, are given by

$$\bar{U}^{(|m|)}_{j'l',jl}(R) = \sum_{k=|l-l'|}^{l+l'} A_k^{(ll'|m|)} \sum_{q=0}^{M} \binom{M}{q} \frac{(-1)^{M+q}}{2^{J+1-q}(J+1-q)!}(\Lambda R)^k \hat{k}_{J-k-q+1/2}(\Lambda R) \tag{3.79}$$

with $M = (l+l'-k)/2$ and now $J = j+j'+l+l'+2$.

3.4.5 Numerical Results

Using the orthogonality property of the rotation matrices, and from visual inspection of rotated 2D and 3D molecular surfaces, it is relatively straightforward to verify that the SH rotation expressions are correct, and that an arbitrary Euler rotation preserves the vector length of the expansion coefficients when working in ordinary C double precision arithmetic.

In order to verify the implementation of the translation matrix calculations, numerical results from the analytic expressions were compared with those from a 2D numerical integration of Eq 3.42 using a regular 200×200 grid in the $(r, \cos\theta)$ plane, and also with a 1D integration of Eq 3.52 using 200 steps in β using log-numerical scheme. Full details are given in Ritchie (2005). This comparison showed that in order to achieve satisfactory numerical precision for all matrix elements up to *N=32* for subsequent calculations in C using ordinary "double precision" instructions (15 decimal digits), it is necessary to perform the calculations using the GMP extended precision library using 160-bit arithmetic for the GTO translation matrices, and using 192-bit arithmetic for the ETO translation matrices. Calculating non-orthogonal expansions was found to be twice as fast as using orthogonal expansions (Ritchie, 2005). However, when translation matrices are needed at a range of regular fixed intervals, it is more efficient to calculate them once in the orthogonal bases and to store them on disc for subsequent use. On the other hand, the non-orthogonal expansions would be more suitable when calculating translations for irregular or unpredictable intervals during a minimisation calculation, for example. In any case, because the rigid body shape matching and docking search spaces can be partitioned into five rotational and one translational degrees of freedom, the analytic expressions for the translation matrices developed here now provide the necessary machinery with which to calculate high order analytic SPF correlations in up to six dimensions.

Chapter 4

Computational Biology Applications

4.1 Molecular Shape Recognition

The term *recognition* is often used to describe the identification of both similar and complementary regions of a pair of proteins or other molecules (Cherfils & Janin, 1993; Dean & Callow, 1987; Katchalski-Katzir *et al.*, 1992; Vakser & Aflalo, 1994; Jones *et al.*, 1995). For a pair of molecules to be considered similar, both their shapes and their physico-chemical characteristics (e.g. electrostatic properties, hydrophobicity, and hydrogen-bonding propensities) must match over the regions of interest. Conversely, these properties must be contraposed appropriately to give a complementary arrangement. Thus the tasks of rotating and translating a pair of molecules to find similar or complementary surface regions may be interpreted as two closely related facets of a common *recognition problem*. If the molecules are assumed to be rigid, then both tasks are characterised by a six-dimensional (6D) search space, as illustrated in Figure 4.1. However, in the similarity case, if both molecules are initially located at the origin, the translational component will often be small and two of the rotation angles, e.g. (β_A, γ_A), will be almost redundant.

4.1.1 SPF Protein Shape Superposition

To demonstrate the SPF correlation technique, this section describes the superposition of a pair of similar proteins, where each protein is represented as a single 3D GTO shape density expansion (Eq 2.125). Letting $A(\underline{r})$ and $B(\underline{r})$ represent the shapes of proteins A and B, respectively, and placing each protein initially at the origin, it is convenient to express the correlation as

$$E \equiv E(\beta_A, \gamma_A, R, \alpha_B, \beta_B, \gamma_B) = \int [\hat{T}_z(-R)\hat{R}(0, \beta_A, \gamma_A)A(\underline{r})][\hat{R}(\alpha_B, \beta_B, \gamma_B)B(\underline{r})]\,\mathrm{d}V, \quad (4.1)$$

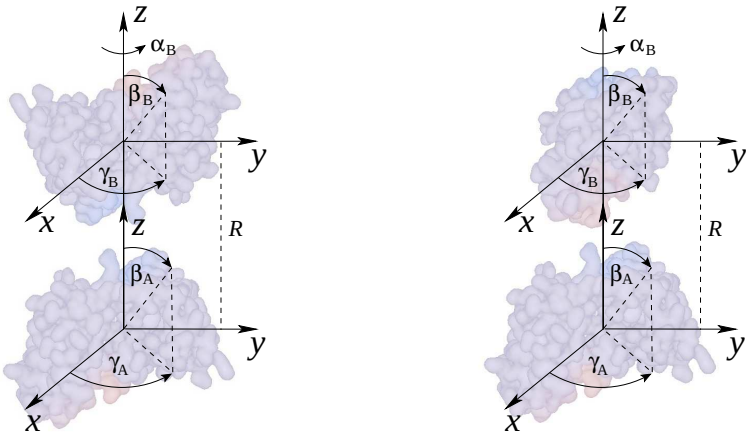

Figure 4.1: Schematic illustration of the 6D rigid body shape-matching (left) and docking (right) search spaces in terms of one translational coordinate, R, and five Euler rotational coordinates, (β_A, γ_A) and $(\alpha_B, \beta_B, \gamma_B)$, assigned to the receptor and ligand, respectively. The distance between the molecular centres will be small and two of the rotation angles will be almost redundant.

where $\hat{T}_z(-R)$ translates the rotated $A(\underline{r})$ by R along the negative z axis. Substituting the SPF representation for each of the terms gives

$$E = \sum_{nln'l'mm'm''}^{N} T^{(|m'|)}_{n'l',nl}(R) R^{(l')}_{mm'}(0, \beta_A, \gamma_A) a_{n'l'm'} R^{(l)}_{mm''}(\alpha_B, \beta_B, \gamma_B) b_{nlm''}. \tag{4.2}$$

Hence the goal is to find the rotational and translational parameters which maximise the above sum. In this and the following sections, the short-hand summation notation used above will be used for brevity, where it is understood from the context that the summation will range over the allowed values of all subscripts, e.g. $|m| \leq l < N$, etc.

Clearly, the cost of calculating directly Eq 4.2 scales in the order of $O(N^7)$ operations for each trial orientation, and it would be prohibitively expensive to calculate this sum repeatedly in an exhaustive 6D search. However, a much more efficient strategy is to compute the sum in stages using pre-calculated rotation and translation matrix elements. For example, assuming all rotation matrix elements have been pre-calculated for a specific (β, γ) rotation, a vector of $O(N^3)$ coefficients for molecule A may be rotated in $O(N^3) \times O(N) = O(N^4)$ operations using

$$a_{nlm}(\beta_A, \gamma_A) = \sum_{m'} R^{(l)}_{mm'}(0, \beta_A, \gamma_A) a_{nlm'}. \tag{4.3}$$

Assuming all translation matrix elements have also been pre-calculated, a vector of coefficients representing a specific (β, γ, R) orientation may be calculated in $O(N^3) \times (O(N) + O(N^2)) = O(N^5)$ operations using

$$a_{nlm}(R, \beta_A, \gamma_A) = \sum_{n'l'} T^{(|m|)}_{nl,n'l'}(-R) a_{n'l'm}(\beta_A, \gamma_A). \tag{4.4}$$

Similarly, vectors of rotated instances of molecule B may be calculated in $O(N^3) \times O(N) = O(N^4)$ operations per orientation using

$$b_{nlm}(\beta_B, \gamma_B) = \sum_{m'} R^{(l)}_{mm'}(0, \beta_B, \gamma_B) b_{nlm'}. \tag{4.5}$$

The final degree of freedom is a twist rotation about the z axis. This could be included in the above formula, but because this rotation may be written as

$$b_{nlm}(\alpha_B, \beta_B, \gamma_B) = \sum_{m'} R^{(l)}_{mm'}(\alpha_B, 0, 0) b_{nlm'}(\beta_B, \gamma_B) \tag{4.6}$$

$$= \sum_{m'} b_{nlm'}(\beta_B, \gamma_B) \cos m'\alpha_B + b_{nl\overline{m}'}(\beta_B, \gamma_B) \sin \overline{m}'\alpha_B, \tag{4.7}$$

it is more efficient to calculate the intermediate quantities

$$P_m = \sum_{nl} a_{nlm}(\beta_A, \gamma_A) b_{nlm}(\beta_B, \gamma_B), \tag{4.8}$$

and

$$Q_m = \sum_{nl} a_{nlm}(\beta_A, \gamma_A) b_{nl\overline{m}}(\beta_B, \gamma_B), \tag{4.9}$$

and to perform the correlation by iterating over all combinations of (R, β_A, γ_A) and (β_B, γ_B) orientations in order to calculate the α_B rotations using a real Fourier series:

$$E = \sum_{m} P_m \cos m\alpha_B + Q_m \sin \overline{m}\alpha_B. \tag{4.10}$$

For high order expansions, the calculation of Eq 4.10 over multiple angular samples, M, may be performed in $O(M \log M)$ operations by using a 1D FFT. However, when *N<16*, it is faster to compute Eq 4.10 for each value of α_B explicitly in $O(MN)$ time. In any case, the overall 6D shape-matching algorithm may be implemented as a nested sequence of transformations, as summarised in Figure 4.2. Despite the relatively high cost of rotating and translating individual 3D coefficient vectors, each distinct (R, β_A, γ_A) orientation of molecule A and each distinct (β_B, γ_B) orientation of molecule B is calculated only once. Thus, the main cost of the superposition search algorithm lies in the $O(N^2)$ combinatorial iteration to compare pairs of transformed A and B vectors, each of which has an inner cycle over the twist angle, α_B.

Figure 4.3 shows some GTO steric density representations of a pair of globular protein domains, the *Streptococcal* pyrogenic exotoxin A1 superantigen (PDB code 1B1Z) (Papageorgiou *et al.*, 1995) and the *Staphylococcus aureus* exotoxin SEC3 (PDB code 1JCK) (Fields *et al.*, 1996). These globular proteins have a relatively low sequence identity of 46% but share a highly similar fold. Hence they may be superposed well by conventional least-squares fitting of conserved C_α coordinates. However,

begin 6D Superposition
  foreach (β_A, γ_A)
    calculate $\underline{a}(\beta_A, \gamma_A)$ using $a_{nlm'}(\beta_A, \gamma_A) = \sum_{m'} R^{(l)}_{mm'}(0, \beta_A, \gamma_A) a_{nlm'}$
  endfor
  foreach (β_B, γ_B)
    calculate $\underline{b}(\beta_B, \gamma_B)$ using $b_{nlm}(\beta_B, \gamma_B) = \sum_{m'} R^{(l)}_{mm'}(0, \beta_B, \gamma_B) b_{nlm'}$
  endfor
  foreach R
    load $T(R)$ from disc
    foreach (β_A, γ_A)
      calculate $\underline{a}'$ using $a'_{nlm} = \sum_{n'l'} T^{(|m)|}_{nl,n'l'}(R) a_{nlm}(\beta_A, \gamma_A)$
      foreach (β_B, γ_B)
        calculate $\underline{P}$ using $P_m = \sum_{m'} a'_{nlm'} b_{nlm'}(\beta_B, \gamma_B)$
        calculate $\underline{Q}$ using $Q_m = \sum_{m'} a'_{nlm'} b_{nl\overline{m'}}(\beta_B, \gamma_B)$
        foreach α_B
          calculate $c[\alpha_B] = \sum_m (P_m \cos m\alpha_B + Q_m \sin \overline{m}\alpha_B)$
        endfor
        save $C[R, \beta_A, \gamma_A, \alpha_B, \beta_B, \gamma_B] = \max_{\alpha_B}(c[\alpha_B])$
      endfor
    endfor
  endfor
end

Figure 4.2: Pseudo-code summary of the 6D superposition search algorithm. Icosahedral tessellations of the sphere are used to generate a near regular pattern of (β, γ) rotational samples for the receptor (A) and ligand (B) rotations, respectively. Vectors of rotated coefficients are calculated once and stored for each rotational sample. The receptor coefficient vectors are then translated for each translational search step, and an inner iteration over the twist angle is performed for each pair of receptor and ligand vectors to complete the 6D search. Arrays of $\cos m\alpha_B$ and $\sin m\alpha_B$ are also pre-calculated outside the main loop. Because the search iterates over a discrete number of rotational and translational steps, the final correlation score may be saved using a single integer identifier (details not shown).

in this illustration the superposition was performed by maximising the overlap between the respective GTO steric density expansions using correlations with *N=6*. A near-identical superposition (not shown) was also achieved by correlating electrostatic charge densities in the ETO basis. From Figure 4.3, it can be seen that the high order GTO expansions capture the detailed shape of each protein remarkably well, although the low order expansions still encode sufficient information to allow a very good global superposition to be calculated. The superposition shown was calculated by searching over some 21×10^6 trial orientations generated from 162 (β, γ) icosahedral angular samples for each protein, 128 twist samples in α_B, and 40 distance steps of 0.25Å. The entire calculation required about 8 seconds on a 2GHz Pentium Xeon processor.

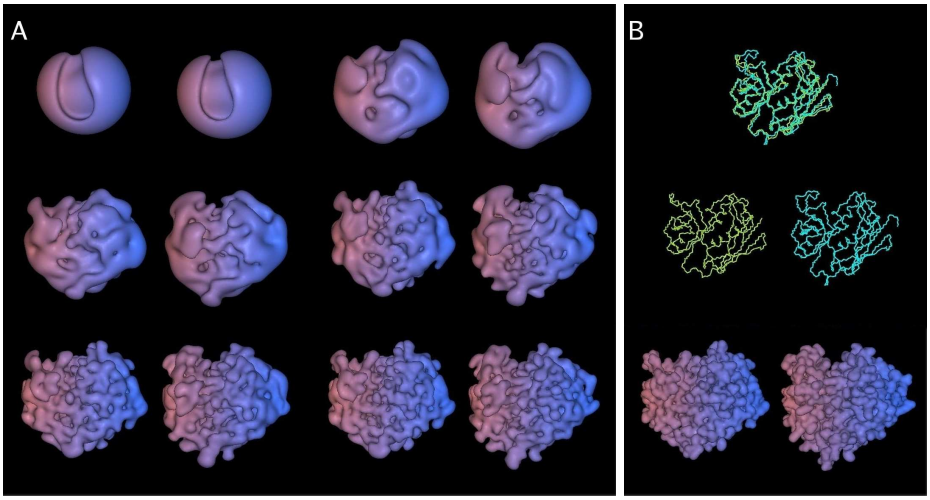

Figure 4.3: Illustration of the GTO shape representation and superposition of a pair of globular proteins, the superantigens SpeA and SEC3. (A) From top left to bottom right: the steric density functions of SpeA and SEC3 shown at expansion orders *N=6, 12, 16, 20, 25,* and *30*. Each pair is in the same superposed orientation, separated horizontally for clarity, with SpeA on the left and SEC3 on the right. For visualisation, each surface shape was contoured from the 3D density function. Expansions to *N=32* are visually almost indistinguishable from the *N=30* expansions, and are not shown here. (B) The corresponding backbone traces of the superposition with SpeA in yellow and SEC3 in cyan (top), the separated backbones with SpeA on the left and SEC3 on the right (centre), and the original molecular surfaces from which the GTO density representations were derived (bottom).

4.1.2 Clustering CATH Protein Structure Super-Families

The BLAST sequence alignment software (Altschul *et al.*, 1990) is probably familiar to all biologists as the standard tool for searching genomic nucleotide or amino acid sequence databases. Biologists often perform sequence alignments as a first step in determining the function and evolutionary origin of an unknown protein or gene sequence. However, this typically requires that there be at least 20% similarity between a known sequence and the query sequence[1]. Nonetheless, proteins can have much lower sequence similarities than this, yet still share the same overall fold. In other words, in Nature, protein folds are more highly conserved than protein sequences. Hence aligning and clustering protein structures can be a useful way to analyse the functional and evolutionary relationships between proteins, even when they have very low sequence similarity (Kolodny *et al.*, 2005). However, whereas symbolic techniques such as the Smith-Waterman and Needleman-Wunsch algorithms for aligning protein sequences have become standard tools in bioinformatics, it remains an open question as to how best to align the 3D structures of proteins (Sippl & Wiederstein, 2008). Most existing structural alignment algorithms are based on comparisons of the C_α backbone traces or vectors formed by the

[1]Proteins with sequence identities in the range 20-35% are sometimes said to be in the twilight zone between similar and non-similar structures (Rost, 1999)

C_α-C_β atoms of each non-glycine amino acid, for example. However, these approaches are significantly more computationally expensive than the symbolic techniques because they typically entail the calculation of multiple least-squares rotation matrices which is expensive in the context of dynamic programming alignment algorithms. For example, current structural alignment algorithms such as e.g. SSM (Taylor & Orengo, 1989), DALI (Holm & Sander, 1991), SAP (Taylor, 1999), CE (Shindyalov & Bourne, 1998), and VAST (Madej *et al.*, 1995), typically consume several seconds of central processor unit time (CPU-seconds) per pair-wise alignment (Kolodny *et al.*, 2005). Hence there is a need to develop fast sequence-independent ways of comparing protein structures.

Within the framework of my ANR Chaires d'Excellence grant at the LORIA, Lazaros Mavridis is currently employed as a postdoctiral assistant on the "3D-Blast" project, which aims to apply 3D SPF techniques to the above task. As a preliminary evaluation of our approach, we performed some alignment and clustering experiments on several proteins selected from the CATH protein structure database (Orengo *et al.*, 1997; Cuff *et al.*, 2008). Lazaros recently presented this work at the 3DSIG satellite meeting of Intelligent Systems in Molecular Biology (ISMB2009-3DSIG), and a paper has been accepted for the publication in the proceedings of the Pacific Symposium on Biocomputing conference (PSB-2010). The rest of this section and the next section summarise the results obtained thus far.

Although Section 4.1.1 described superposing a pair of proteins by maximising the overlap between their van der Waals volumes, it is convenient to use a normalised scoring function when comparing multiple proteins. Hence the following calculations use a Carbo similarity score (Carbo *et al.*, 1980) calculated as:

$$S = \frac{\underline{a}.\underline{b}'}{|\underline{a}|.|\underline{b}|}, \tag{4.11}$$

where $\underline{b}'$ denotes a vector of rotated shape-density coefficients of molecule B, $|\underline{b}|$ denotes the magnitude of that vector, and the inner product notation is short-hand for

$$\underline{a}.\underline{b}' = \sum_{nlm}^{N} a_{nlm} b'_{nlm}, \tag{4.12}$$

This cosine-like scoring function gives values that range from zero (no similarity) to unity (two identical proteins in perfect alignment).

In the CATH classification, proteins are assigned to a super-family according to their fold class, architecture, topology, and homology. This is essentially a hierarchical scheme, with the top-level class consisting of four possible fold types: All-α (i.e. the protein structure consists entirely of α-helical secondary structure elements), All-β (the protein structure consists entirely of β-sheet secondary structure elements), α+β (the structure contains both α-helices and β-sheets), and "irregular" (no identifiable secondary structure elements). Each of the four levels in the CATH hierarchy is identified by a numeric code. Additionally, CATH names each protein according to its four-letter PDB code, its

chain letter, and the number of domains (e.g. 1IOMA02). For each clustering experiment, five or six super-families with the same architecture were selected in such a way as to give around 30 protein structures for each CATH fold class. Hence the aim of these experiments is to assess how well our approach can identify proteins with the same topology and homology within a given fold architecture. The details of the CATH super-families used here are shown in Table 4.1.

Table 4.1: The 23 CATH super-families used in the protein clustering experiments.

Class + Architecture	Topology + Homology	Protein Name and Function	Representative Structure
All-α Orthogonal Bundle (1.10)	230.10	Citrate synthase	1iomA02
	120.10	Trypsin/Alpha-Amylase Inhibitor	1beaA00
	225.10	NK - Lysin	1l9lA00
	167.10	G-Protein Signalling Regulator	1dk8A02
	30.10	HMG DNA Binding Domain	1qrvA00
All-β Ribbon (2.10)	109.10	LexA repressor	1jhfB00
	150.10	Urease	1ejxB00
	110.10	LIM domain PINCH protein	1g47A00
	77.10	Hemagglutinin	1jsdA02
	160.10	Endothelial Growth Factor 165	1kmxA00
	10.10	PDC-109	1h8pA02
α+β Roll (3.10)	130.10	Ribonuclease A	1dy5A00
	170.10	Elastase	1u4gA01
	150.10	DNA Polymerase	1ok7A01
	110.10	Ubiquitin Conjugating Enzyme	2grrA00
	120.10	Flavocytochrome B2	1cyoA00
Irregular (4.10)	280.10	MYOD Helix-Loop-Helix Domain	1nlwE00
	290.10	Bacteriorhodopsin Fragment	1bctA00
	410.10	Factor Xa Inhibitor	1g6xA00
	490.10	HiPIP	1iuaA00
	400.10	LDLR	2fcwB02
	320.10	Dihydrolipoamide Transferase	1w85I00

For the All-α class, five CATH super-families were selected as listed in Table 4.1. For each pair of proteins in this set, a correlation search was performed to find the orientation that gives the maximum Carbo similarity score (Eq 4.11). Ward's agglomerative clustering (Ward, 1963) was then applied to the resulting table of pair-wise similarity scores to give a total of five clusters. The clustering results in Figure 4.4 show that the 1.10.230.10 and 1.10.167.10 super-families are correctly assigned to two separate groups. Although there exist some difference in the clusters produced for the remaining super-families, it can be seen that there is still a very good agreement between the calculated SPF clusters and the CATH hierarchy. The most notable exception is that suposin (PDB code 1N69) is grouped with the 1.10.30.10 super-family. From visual inspection of Figure 4.4 it can be seen that the overall fold of suposin is much closer to that of the 1.10.30.10 super-family than the CATH assignment

of 1.10.225.10. This suggests that the automatic SPF classification could potentially help the CATH curators resolve unusual or ambiguous cases.

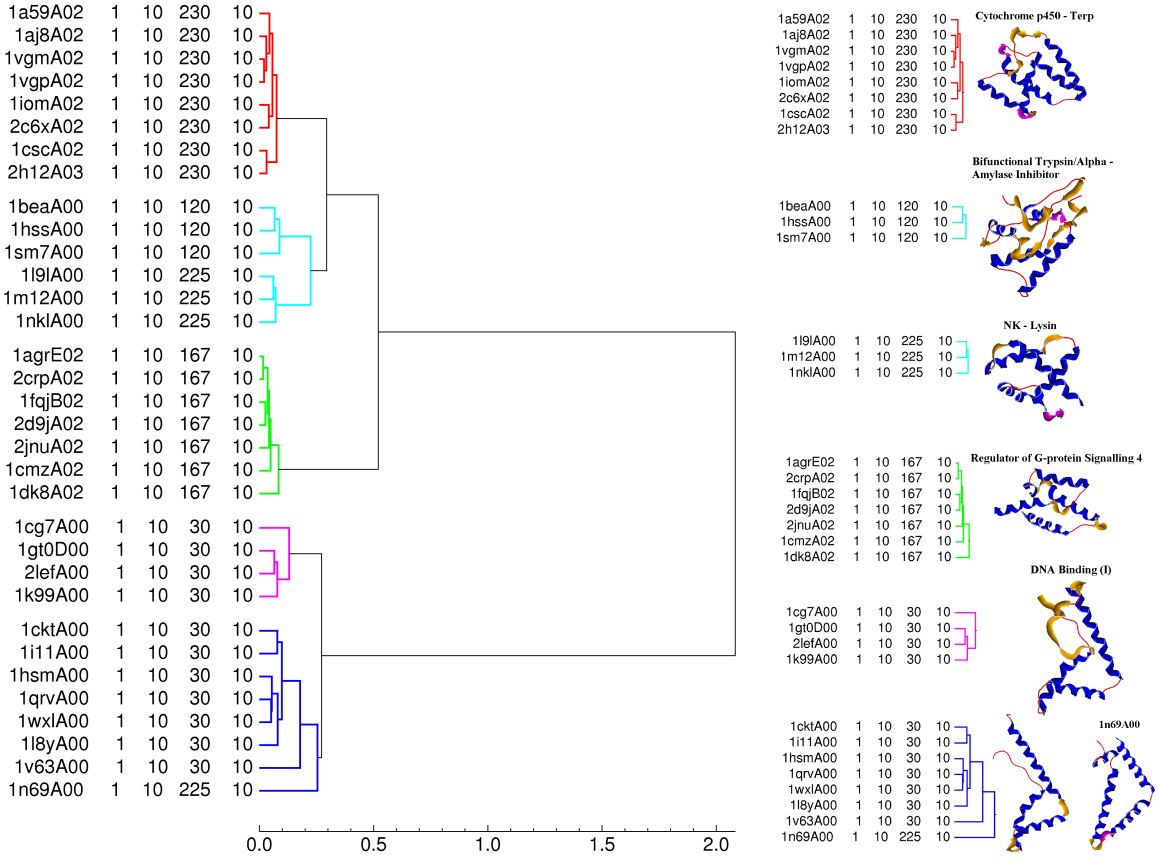

Figure 4.4: SPF clustering results of the All-α class using *N=6* correlations and requesting five clusters.

For the All-β class, six super-families were clustered. As can be seen in Figure 4.5, SPF clustering correctly distinguishes all six groups, but two proteins are mis-placed according to the CATH classification. These are the carboxy-terminal LIM domain (PDB code 1CTL) and the influenza virus haemagglutinin (PDB code 2VIR) which are grouped with the singleton heparin-binding domain (PDB code 1KMX). This seems to occur because 1CTL and 2VIR are calculated to be less similar to the other members of their respective CATH super-families, and they are grouped with 1KMX largely because all three proteins have similar steric bulk.

For the α+β class, five CATH super-families were clustered. Figure 4.6 shows that the 3.10.130.10 and 3.10.120.10 super-families are correctly assigned into two groups. The remaining three super-families (3.10.110.10, 3.10.150.10, and 3.10.170.10), present a similar case to the All-β results whereby one super-family group (3.10.110.10) is split into two, and two super-family groups (3.10.170.10 and 3.10.150.10) are merged into one. Nonetheless, despite these differences the overall consistency

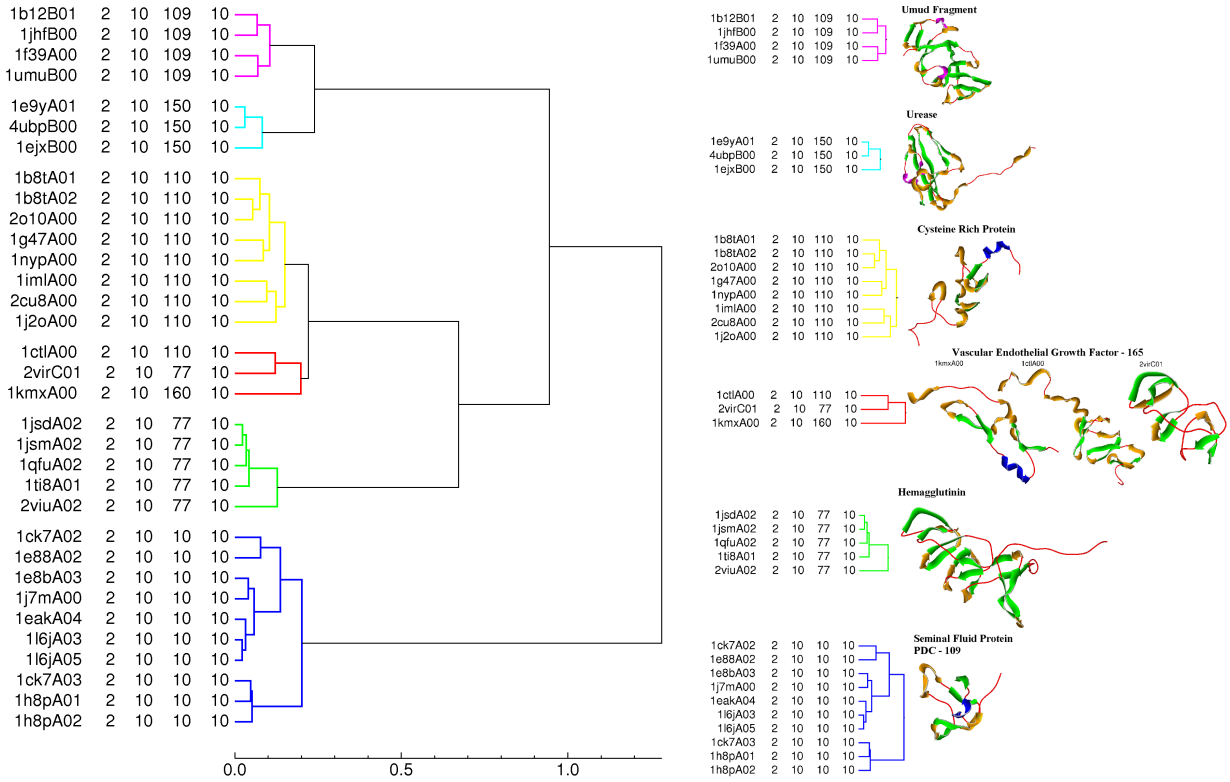

Figure 4.5: SPF clustering results of the All-β class using *N=6* with six clusters.

of the SPF clustering with the CATH hierarchy is clearly very good.

For the irregular class, six super-families were clustered. Like the All-β example, SPF clustering is completely consistent with the CATH hierarchy. However, two proteins are misplaced with respect to the CATH classification, namely bikunin from the inter-alpha-inhibitor complex (PDB code 1BIK) and the tick anticoagulant protein (PDB code 1D0D), which are grouped with the 4.10.490.10 super-family. This seems to be due to the difference in size between those proteins and the rest of their super-family of factor Xa inhibitors. For example, Figure 4.7 shows that bikunin has a repeat of the same motif as the other factor Xa inhibitors. Hence it is sterically too large to be clustered with the other Xa inhibitors, and is instead placed with the larger proteins of the 4.10.490.10 super-family.

4.1.3 Searching the CATH Protein Structure Database

As a second test of the utility of the approach, the entire CATH database of some 12,000 structures was searched using SPF density functions as queries. When searching such a large database for protein structures which match a given query, it would be desirable to be able to eliminate rapidly many database proteins which have substantially different shapes from the query and which cannot

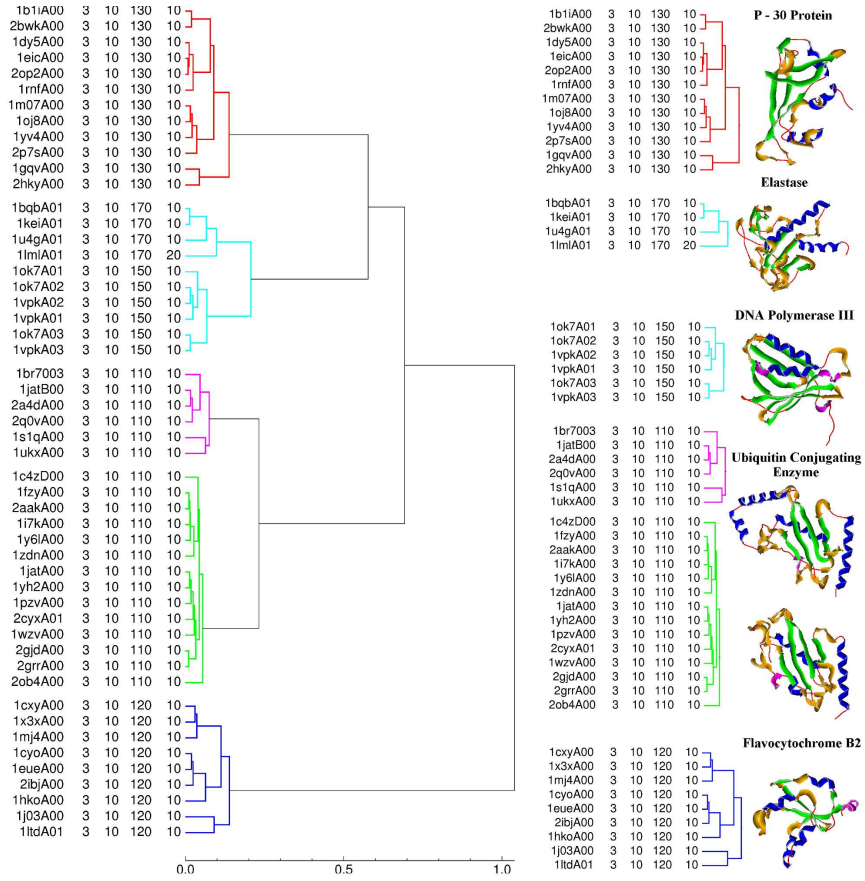

Figure 4.6: SPF clustering results of the α+β class using *N=6* with five clusters.

possibly overlay it well in any orientation. Noting that expansion coefficients with the same values of m transform amongst themselves under rotation, it is natural to use the vector interpretation of SH coefficients to construct rotationally invariant (RI) fingerprints (RIFs) as:

$$A_n = \sum_{l=0}^{n-1} \sum_{m=-l}^{m=l} a_{nlm}^2. \tag{4.13}$$

If the coefficients a_{nlm} define the shape density of a protein, then the rotation-invariant descriptors A_n together encode the protein's radial mass distribution. By analogy to Eq 4.11, the RIF similarity score is written as:

$$S_{RIF} = \frac{\sum_{n=1}^{N} A_n B_n}{\left(\sum_{n=1}^{N} A_n^2\right)^{1/2} \left(\sum_{n=1}^{N} B_n^2\right)^{1/2}}. \tag{4.14}$$

For the initial database search experiment, asparagine synthetase (PDB code 12AS, CATH super-family 3.30.930.10) was selected as the query structure. The 3.30.930.10 super-family has 27 mem-

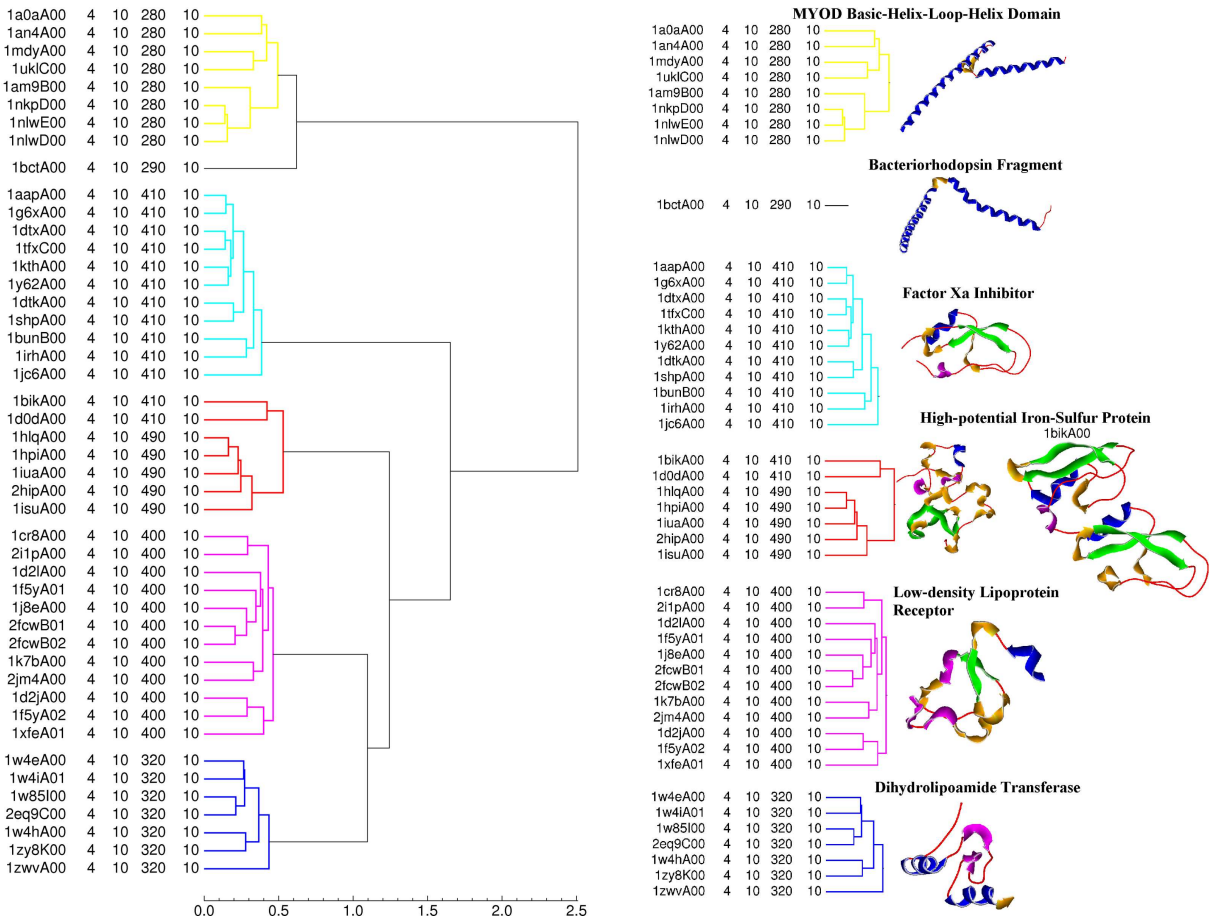

Figure 4.7: SPF clustering results of the Irregular class using *N=6* with six clusters.

bers, and these were treated as "positives" while the remaining proteins in the database were treated as "negatives" with respect to the query. If a scoring function were to reproduce exactly the CATH classification, the 27 positives would appear at the top of the ranked list. However, such an ideal outcome is seldom observed in practice. Therefore, Receiver-Operator-Characteristic (ROC) curves (Egan, 1975; Fawcett, 2006) are used to analyze objectively the precision/recall characteristics of the scoring functions. In a ROC analysis, each element of the ranked list is considered in turn, and the number of positives and negatives in the sublists to each side of the current element are counted. Here, the high similarity sublist is called the "hit list". A true positive (TP) is assigned when an element in the hit list contains an original positive, and a false positive (FP) is assigned if that element contains a negative. Conversely, a true negative (TN) is assigned when an original negative falls outside the hit list, and a false negative (FN) is assigned if that position is occupied by a positive member. ROC plots are produced by plotting the true positive rate (TPR) on the y axis against the false positive rate

(FPR) on the x axis, where TPR and FPR are given by:

$$TPR = \frac{TP}{TP + FN} \tag{4.15}$$

and

$$FPR = \frac{FP}{FP + TN}. \tag{4.16}$$

The quality of a scoring function can quickly be assessed from the shape of a ROC plot. For example, a random scoring function would give a diagonal line (TPR=FPR), whereas a perfect scoring function would give a horizontal line (TPR=1) with a maximum value for the area under the curve (AUC=1).

Because it is relatively time-consuming to calculate high order correlations, different rotational and rotationally invariant parameters were tested to explore the extent to which queries may be performed more rapidly without sacrificing accuracy. In each case, the asparagine synthetase structure (PDB code 12AS, CATH super-family 3.30.930.10) was superposed onto each protein in the database using *N=6* SPF correlation searches, and the database structures were ranked in order to similarity to the query. Figure 4.8 shows the resulting ROC curves obtained for a range of expansion orders when querying the CATH database. This figure shows that the overall approach gives very good precision and recall for all expansion orders above *N=2*, and that there is very little benefit in using expansion orders greater than *N=6*. Figure 4.9 shows the 27 members of this super-family which were treated as TPs with respect to the query. To analyze the results further, the TPs were clustered into 5 groups. The query belongs to Group 1, and all members of this group were found in the top 10 hits when N$\geq$6. Groups 2 and 3 have similar β-sheet structures to Group 1, but different arrangements of α-helices. All proteins in Groups 2 and 3 are ranked in the top 20% of the database, and all proteins in Group 4 are ranked in the top 30%. Finally, the singleton Group 5 is an obvious outlier due to its extra α-helical domain.

Figure 4.8 also shows the results for the RIF scoring function. Compared to the rotation-dependent scoring function, the RIF function generally performs remarkably well. However, the two functions behave rather differently on the first percentages of the database. For example, the rotational searches give a TPR of around 40% on the first 0.1% of the database, whereas the RIF searches give a TPR of only around 10%. Hence the RIF function is not sufficiently sensitive to be used on its own but it could usefully be used as a fast pre-filter on the database so that the more expensive rotation-dependent function need only be applied to the most promising candidates.

In order to test the above notion, the CATH database was searched using the RIF and rotational scoring functions in tandem using several protein structures as queries: asparagine synthetase, ALF4-activated Giα1 protein (PDB code 1AGR), chicken cysteine-rich protein (PDB code 1B8T), dihydrolipoyllysine-residue acetyl transferase (PDB code 1W4E), and UbcH7 (PDB code 1C4Z). Using a RIF pre-filter similarity threshold of 0.99, which selects from 2% to 15% of the database for rotational re-scoring, each tandem search takes less than 10 minutes compared to 75 minutes for full rotational

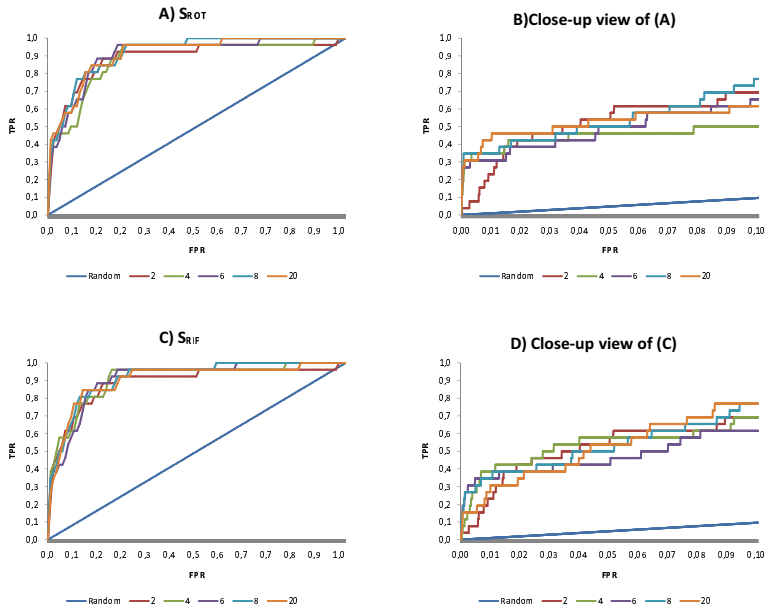

Figure 4.8: ROC plot analyses obtained when querying the entire CATH database with the 12asA00 structure. A: rotation-dependent (ROT) scoring function, Eq 4.11; B: close-up view of (A); C: RIF scoring function,Eq 4.14; D: close up view of (C).

searches on a 2.3GHz Pentium Xeon processor. Figure 4.10 shows the AUCs for the top 1% of the database for the rotational, RIF, and the tandem searches. This figure shows that tandem searches achieve the same high level of performance as the rotational searches. Table 4.2 presents the overall AUCs obtained for these searches. The very high values in this table confirm the utility of the SPF scoring functions.

Table 4.2: Summary of AUC values obtained when searching the entire CATH database.

Query	RIF	ROT	RIF+ROT
12AS	0.944	0.907	0.929
1AGR	0.960	1.000	1.000
1B8T	0.964	0.983	0.997
1W4E	0.995	0.999	0.997
1C4Z	0.968	0.995	0.995

This table lists the AUC values obtained when searching the entire CATH database with the given protein structures (listed by PDB code) as queries for the RIF, rotational (ROT), and tandem (RIF+ROT) searches.

Overall, the above results show that low resolution SPF expansions provide a reliable and fast sequence-independent way to superpose and compare protein structures. We believe that the SPF approach could provide an automatic and objective way to enhance the quality of protein structure classifications, and we are working to provide a web interface for real-time protein structure queries.

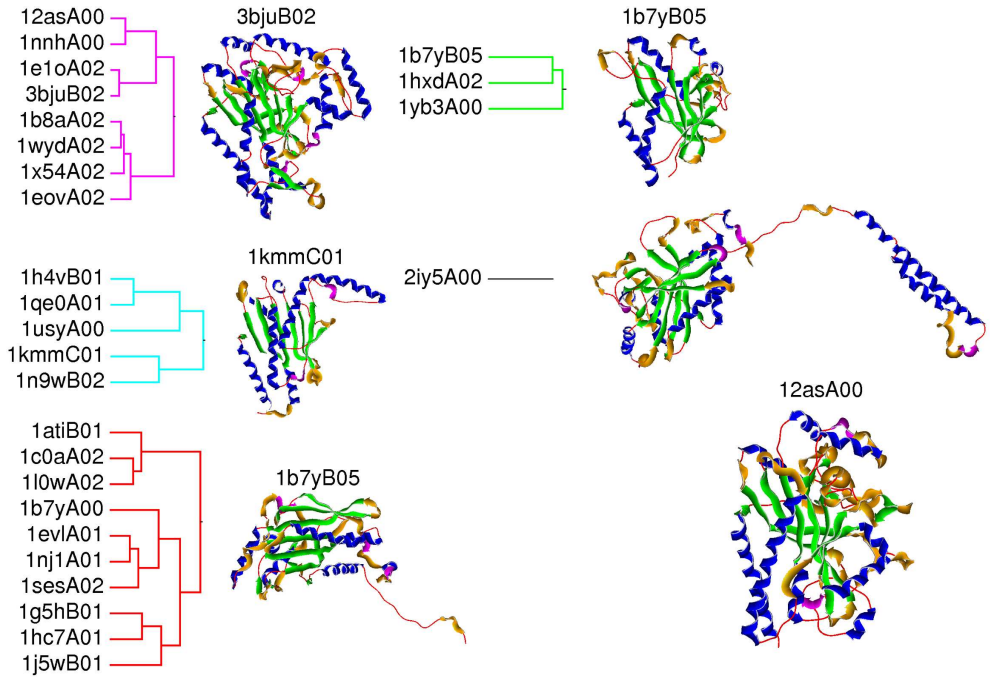

Figure 4.9: Clustering the 3.30.930.10 super-family into five groups. The representative member of each group is shown along with the query protein 12asA00.

4.2 SPF Protein-Protein Docking

In order to express protein *shape complementarity* in a suitable form for calculating SPF docking correlations, it is useful to define a surface skin function, $\sigma(\underline{r})$, as a shape density function which describes the volume bounded by the VDW surface and the SAS surface of each protein. In other words,

$$\sigma(\underline{r}) = \begin{cases} 1 & \text{if } \underline{r} \in \text{ surface skin,} \\ 0 & \text{otherwise.} \end{cases} \tag{4.17}$$

This density function may be calculated numerically on a grid in a similar way to the calculation of the VDW density function, $\tau(\underline{r})$. The use of a surface skin representation to model protein shape complementarity is justified by considering Figure 4.11. This figure suggests that a good strategy for finding complementary orientations between a pair of proteins is to maximise the overlap between the interior shape density of one protein with the exterior skin region of the other. Steric clashes may be penalised with an interior-interior shape-density overlap penalty term. Using these ideas, the shape complementarity score, E, for proteins A and B may be written as

$$E = \int \sigma_A \tau_B \mathrm{d}V + \int \tau_A \sigma_B \mathrm{d}V - \int Q \tau_A \tau_B \mathrm{d}V, \tag{4.18}$$

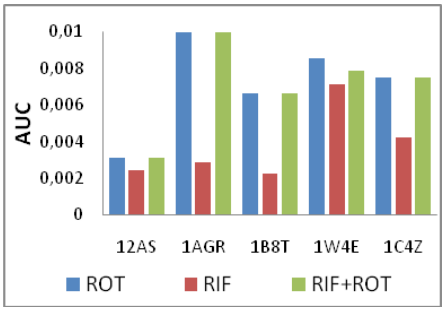

Figure 4.10: ROC plot AUCs obtained when searching the entire CATH database using the five selected query proteins, but considering only the top 1% of the hits found by the rotational (ROT), RIF, and tandem (RIF+ROT) scoring functions.

where $\sigma_A \equiv \sigma_A(\underline{r}_A)$, etc., and where Q is a positive interior-interior penalty factor. I currently use $Q = 11$. When the SAS is calculated using a 1.4Å probe radius, the first two terms give an expression for the volume of water expelled from the protein surfaces upon association (see Figure 4.11). With a suitable scale factor, this expelled volume can be used as a first-order approximation to the hydrophobic free energy of association (Richmond, 1984). By putting $Q_B = \sigma_B - Q\tau_B$, Eq 4.18 can be written as a two-term pseudo-energy expression

$$E_{\text{SHAPE}} = K \int \left(\sigma_A \tau_B + \tau_A Q_B\right) \mathrm{d}V, \tag{4.19}$$

where K is a negative constant which gives negative scores to favourable orientations.

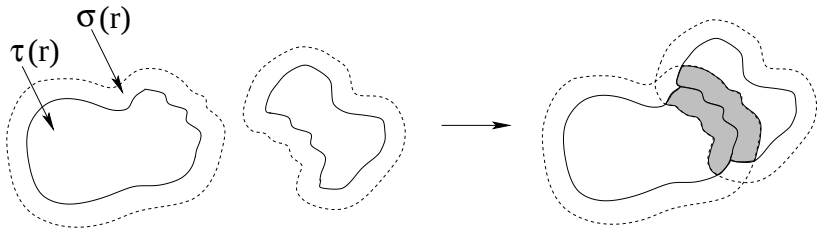

Figure 4.11: Schematic illustration of shape complementarity using 3D density functions. Here, solid lines represent the VDW surfaces, and dashed lines represent the SASs. The function $\tau(\underline{r})$ is defined as unity inside the VDW surface and zero everywhere else; $\sigma(\underline{r})$ is defined as unity within the volume bounded by the SAS and the VDW surface, and zero everywhere else. When a pair of proteins are brought together in a complementary arrangement, there is a large overlap (shaded region) between $\sigma(\underline{r})$ of one protein and $\tau(\underline{r})$ of the other, and *vice-versa*.

4.2.1 Protein-Protein Docking using 1D FFTs

From Section 2.5.2, the *in vacuo* electrostatic interaction energy of a pair of proteins with charge densities $\rho_A(\underline{r})$ and $\rho_B(\underline{r})$ and with electrostatic potentials $\phi_A(\underline{r})$ and $\phi_B(\underline{r})$, respectively, is given by

$$E_{\text{ELEC}} = \frac{1}{2}\int \big(\rho_A(\underline{r})\phi_B(\underline{r}) + \rho_B(\underline{r})\phi_A(\underline{r})\big)\mathrm{d}V. \tag{4.20}$$

Therefore, representing each function as a spherical polar ETO expansion at the origin using the $S(r)$ radial functions, Eq 2.101 immediately gives the electrostatic correlation expression

$$E_{\text{ELEC}}(R,\beta_A,\gamma_A,\alpha_B,\beta_B,\gamma_B) = \frac{1}{2}\sum_{nlm}^{N}\big(a'^{\rho}_{nlm}b'^{\phi}_{n'l'm'} + a'^{\phi}_{nlm}b'^{\rho}_{n'l'm'}\big), \tag{4.21}$$

where a'^{ρ}_{nlm} and a'^{ϕ}_{nlm} denote rotated and translated expansion coefficients of the charge density and electrostatic potential of protein A, etc. according to Equations 4.3–4.6.

Similarly, from Eq 4.18, the shape-density pseudo-energy may be expanded as

$$E_{\text{SHAPE}}(R,\beta_A,\gamma_A,\alpha_B,\beta_B,\gamma_B) = K\sum_{nlm}^{N}\big(a'^{\sigma}_{nlm}b'^{\tau}_{n'l'm'} + a'^{\tau}_{nlm}b'^{Q}_{n'l'm'}\big). \tag{4.22}$$

Hence an overall shape plus electrostatic pseudo-energy for the system can be calculated as

$$E_{\text{TOTAL}}(R,\beta_A,\gamma_A,\alpha_B,\beta_B,\gamma_B) = E_{\text{ELEC}}(R,\beta_A,\gamma_A,\alpha_B,\beta_B,\gamma_B) + E_{\text{SHAPE}}(R,\beta_A,\gamma_A,\alpha_B,\beta_B,\gamma_B). \tag{4.23}$$

This docking correlation expression has been implemented as a nested series of 1D FFTs, as described in Section 4.1.1. In order to obtain satisfactory docking results it is necessary to use (β,γ) angular search increments of around 7.5° generated from icosahedral tessellations of 812 vertices, and to use SPF expansions to at least order *N=25*. However, because the expansion order may be varied independently of the angular search step sizes, the calculation can be accelerated significantly by performing a low resolution shape-only scan of the search space using *N=16*, and by re-scoring only the best 20,000 orientations using high order *N=25* shape plus electrostatic correlations, for example. The results obtained using this approach on a number of protein-protein complexes have been published (Ritchie & Kemp, 2000). When electrostatic interactions are important in a particular complex, the electrostatic term can often help to boost the score of orientations that resemble the correct solution. However, in some cases, the electrostatic contribution worsens the quality of the predictions. Unfortunately, it is not always obvious whether electrostatics should be included in any particular case. In essentially all cases studied, shape complementarity is found to be considerably more important than electrostatics. In Eq 4.18, I set K = -0.6 KJ/mol/Å^3, which typically makes the shape component about five times larger than the electrostatic contribution to the overall score. Nonetheless, the above approach was also used successfully on several of the docking targets in the

CAPRI blind docking experiment (Ritchie, 2003; Mustard & Ritchie, 2005). Figure 4.12 shows the best orientations achieved by *Hex* for two of the blind docking targets in the first rounds of the CAPRI blind docking experiment.

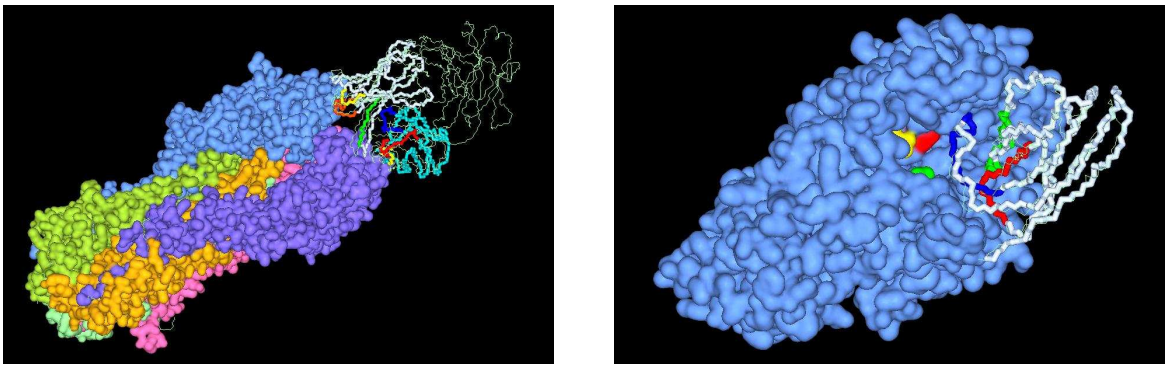

Figure 4.12: Blind docking result for CAPRI Targets 3, HA/HC63 (left), and 6, α-Amylase/AMD9 (right). The best docking solution obtained for the HA/HC63 complex was the 4th solution submitted to CAPRI. This solution has 43/63 correct residue contacts. The deviation between the coordinates of the antibody Fv fragment and those of the crystal structure of the complex is 7.43Å RMS. The HC63 Fv fragment is coloured as VH: white; VL: cyan; H1: orange; H2: yellow; H3: green; L1: red; L2: yellow; L3: blue. The crystallographic orientation of the Fv is shown in light green. The HA chains are coloured as A: light blue; C: pink; E: dark blue; F: orange. The docking solution obtained for the AMB9/α-amylase complex (right) was the 5th solution submitted. This prediction has 53/65 correct residue contacts and a deviation of 2.16Å RMS between the predicted and actual AMB9 coordinates. The α-amylase is in blue and the AMB9 VHH domain is in white. The active-site amylase residues are coloured ASP-197: red; GLU-233: yellow; ASP-300: green. The VHH CDR loops are coloured as CRD1: red; CDR2: green; CDR3: blue. The crystallographic orientation of the VHH is shown in light green.

4.2.2 Focusing Docking Correlations

Compared to conventional Cartesian grid FFT approaches, in the SPF approach it is relatively straightforward to use prior knowledge about a protein-protein interaction to focus and accelerate the docking calculation. For example, if it is known that even just one residue from each protein is involved in the interaction, this is sufficient to set up an initial docking orientation and to restrict or focus the docking search around the initial interface. This is illustrated in Figure 4.13. Using this approach to constrain the search space can significantly enhance the quality of docking predictions (Ritchie *et al.*, 2008).

4.2.3 Docking Very Large Proteins

Because the radial basis functions fall off rapidly beyond about 30Å from the chosen origin, the above approach is not directly suitable for docking very large proteins, such as CAPRI Targets 2 and 3. However, it is not necessary to rely on a single coordinate origin. For example, the surface of the larger “receptor” protein may be covered with multiple copies of the smaller “ligand” protein, and a

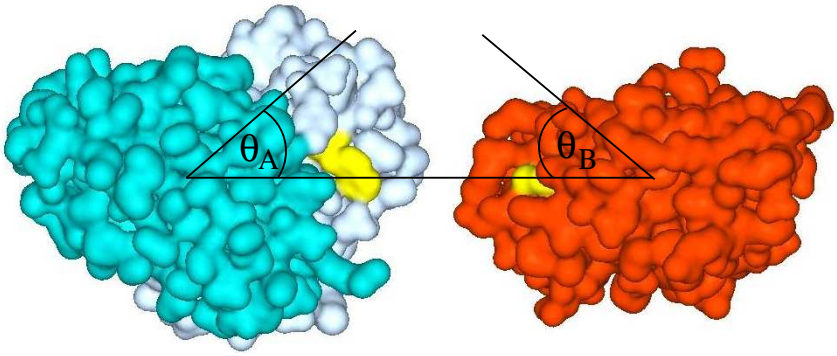

Figure 4.13: Illustration of a focused docking search using prior knowledge of one interaction residue on each docking partner. This figure shows the starting orientation for the HyHel-5 antibody/lysozyme complex, in which the antibody H-33:TRP (left) and the lysozyme Y-53:TYR (right) residues have been highlighted in yellow. The orientation shown was set up automatically in *Hex* by rotating the C_α atom of each highlighted residue onto the intermolecular z axis. The docking correlation may then be constrained to search around the initial orientation by restricting the allowed ranges for the θ_A and θ_B search angles.

focused docking search may be performed around each initial position of the ligand. Figure 4.14 illustrates this approach for CAPRI Target 2, a complex between the large VP6 viral surface protein and an antibody Fv domain.

The docking covering algorithm is as follows: First, if knowledge of the ligand binding surface is available, the ligand molecule is oriented along the negative z axis to face the receptor. Second, a low resolution *L=5* spherical harmonic surface is calculated for the receptor by sampling its surface onto an icosahedral tessellation of the sphere, as shown in Figure 4.14(B). At each triangular facet of the surface, a normal vector is calculated and a 15Å radius sphere is centred on each outward normal, tangential to the surface. This covers the surface with spheres. In the third step, the surface spheres are culled by iteratively identifying and striking out that sphere which has the greatest volume overlap with its neighbours. This procedure is repeated until no overlap volume exceeds 5Å^3. This yields a fairly even distribution of the surviving spheres over the surface of the receptor. Finally, each surviving sphere (normal vector) is used to define a local intermolecular axis for docking, with the initial axis (ligand) orientation being transferred onto the outward normal, and a local coordinate origin for the receptor being defined at an equal distance along the inward normal. Figure 4.14(C) shows the result of applying the surface spheres algorithm to only the C chain of the VP6 trimer, and Figure 4.14(D) shows the trimer covered with 23 generated MCV Fv starting orientations.

4.2.4 Clustering Docking Solutions

Because the macromolecular surface sphere covering procedure tends to over-sample the orientational search space, all low energy solutions are clustered in order to identify distinct orientations.

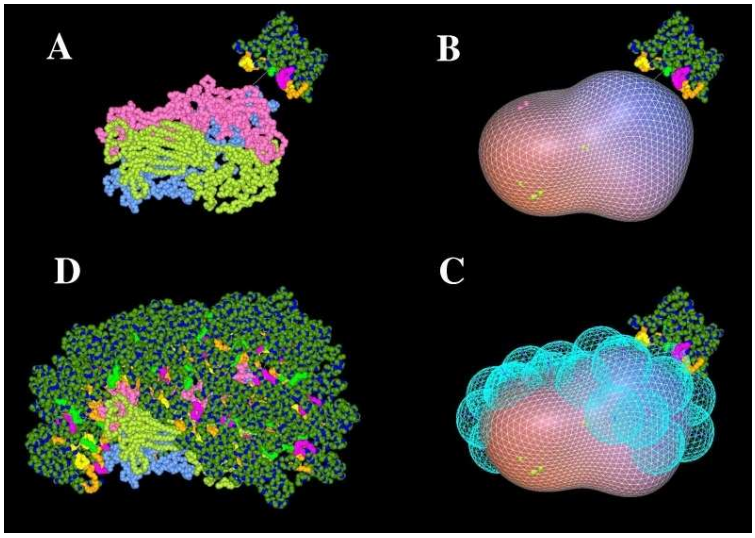

Figure 4.14: The four stages of the macromolecular surface sampling algorithm, illustrated schematically for the antibody MCV/VP6 complex (CAPRI target 2). (A) The hypervariable loops of the MCV Fv fragment (the "ligand") are initially oriented to face the VP6 trimer (the "receptor"). The VP6 chains are coloured as A: blue; B: yellow; C: pink. (B) A low resolution (L=5) spherical harmonic surface is calculated for the receptor (2,252 surface triangles). (C) The spherical harmonic receptor surface after applying the sphere covering algorithm to the C chain of the receptor. (D) Multiple initial docking orientations for the ligand are generated from the sphere centres. This example shows 23 MCV Fv fragments distributed over the VP6 C chain.

The clustering algorithm first orders the docking solutions by energy, and allocates the lowest energy solution as the "seed" member of the first cluster. The list of remaining solutions is then scanned for unallocated entries, and any orientations for which the ligand C_α atoms fall within 2Å RMS of the corresponding atoms in the seed member are allocated to the current cluster. The list is then re-scanned for the next unallocated low energy solution which becomes the next cluster's seed, and the procedure is repeated until all solutions have been allocated to a cluster.

Even when it is not necessary to use multiple ligand starting orientations, this clustering algorithm provides a useful way to reduce the number of "false-positives" generated by a docking search. For example, in an exhaustive search such as ours, many similar but nonetheless distinct orientations may be found, and these would tend to "push a good solution down the list" if clustering were not used. Clustering is also useful when using energy minimisation because multiple docking solutions can coalesce to a single minimised orientation.

4.2.5 Protein Docking Using PCA-Selected Probe Potentials

One of the difficulties in macromolecular docking is to devise a reliable energy-based scoring function with which to evaluate trial docking orientations. Shape complementarity is very effective as an initial filter, but it would be desirable to incorporate chemical interactions in the scoring scheme to help

distinguish the true complex from the many false-positives generated by a docking correlation search. However, there is no direct way to infer which of the several types of intermolecular interactions (electrostatics, hydrogen bonding, desolvation, salt bridges, dispersion forces, etc.) provide the driving force for binding in any particular case. Therefore, as part of the PhD thesis project of Alessandra Fano, we investigated the use of principal component analysis (PCA) of Molecular Interaction Fields (MIFs) to select the most significant types of chemical interaction for a given docking target. We tested this approach by attempting to dock the unbound components of the complex between streptomyces subtilisin (here called SUP) and its natural inhibitor (SSI). This complex provides a good test because it is known to be a difficult case using current techniques.

First, we used the GRID program (Goodford, 1985) to generate 3D potential energy maps for several types of probe atom placed in a grid about the surfaces of each protein. Only points with potential values above a given threshold were retained; all other points were discarded. Figure 4.15 illustrates this approach for the SSI protein. A similar calculation was performed for SUP. Although the aim is to match distributions of such potentials with complementary distributions of potentials on the docking partner, it would be impractical to use all possible potential energy maps during a docking calculation. Hence, we used PCA to select the most relevant and significant types of probes to guide the docking search. PCA is a standard chemoinformatics approach to extract information from very large data sets. PCA is commonly used in small-molecule 3D quantitative structure-activity relationship (3D-QSAR) drug design studies (Pastor & Cruciani, 1995), and it has also been used to map receptor-ligand binding sites (Matter & Schwab, 1999). However, Alessandra's approach is the first time that PCA of probe potentials has been used to score macromolecular docking. The main reason for performing a PCA on the MIFs is that those probes which contribute most to the variance in the energy maps should be expected to be the ones which are most likely to be the best indicators of complementarity for the system under consideration.

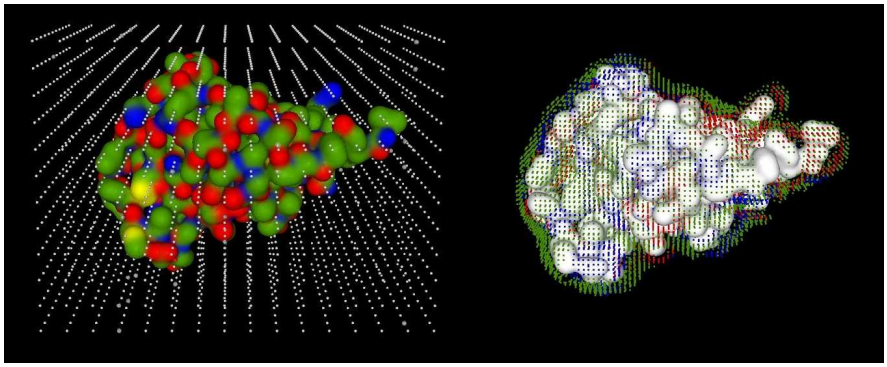

Figure 4.15: Left: the SSI protein placed within a 3D grid. Right: surface probe positions colour-coded by GRID potential – red: proton donor; blue: proton acceptor; green: hydrophobic.

In a PCA, the matrix of probe potentials is decomposed in two smaller matrices of loadings and

scores, respectively. The loadings measure the weight of the original variables in the analysis, and the scores give a simplified picture of the objects (the probes in our case) in terms of a small number of new uncorrelated variables (the principal components, or PCs). Plotting the object scores against the PCs allows those objects (or clusters of objects) that explain most of the variance to be identified. The PCA plots shown in Figure 4.16 show the distribution of six most significant probes (C sp3, NH amide, N+ sp3, carbonyl O, and carboxyl O-) in the first two components of the chemometric space for SUP and SSI. It is interesting to note the mirror-like symmetry in the horizontal plane. This suggests good chemical complementarity between the proteins. These plots show that only three probes (N+, O-, and Dry) are sufficient to explain most of the variance in the potential maps. Hence only those three probes were used in the subsequent docking calculations.

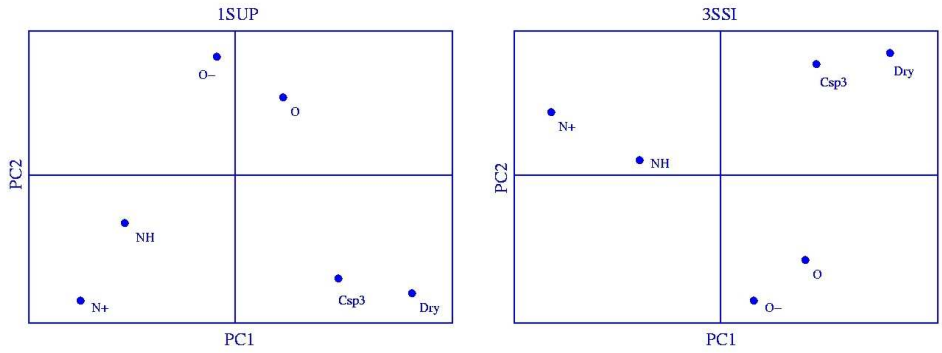

Figure 4.16: Left: the PCA plot for SUP showing the contributions of the six most significant probe potential types (C sp3, NH amide, N+ sp3, carbonyl O, carboxyl O-, and Dry) to the first two principal components, PC1 and PC2. PC1 and PC2 explain 69.3% and 21.5% of the total variance, respectively. Right: similarly, the PCA plot for SSI. PC1 and PC2 explain 74.3% and 16.0% of the total variance, respectively.

For any given probe type, the PCA-selected probe potentials may be transformed into smooth continuous functions for docking within *Hex* in the same way that point charges are transformed to give a charge density function (see Section 2.5.2). For example, by treating the N+ probe positions for protein A as a sum over point potentials, $\phi^{N+}(\underline{x}_i)$, one can write the total potential as:

$$\phi^{N+}(\underline{x}) = \sum_i \phi^{N+}(\underline{x}_i)\delta(\underline{x} - \underline{x}_i). \tag{4.24}$$

This may then be represented as a truncated SPF series in the usual way

$$\phi^{N+}(\underline{r}) \simeq \sum_{nlm} a^{N+}_{nlm} R_{nl}(r) y_{lm}(\theta, \phi), \tag{4.25}$$

where the expansion coefficients are calculated using (c.f. Eq 2.133):

$$a^{N+}_{nlm} = \sum_i \phi^{N+}(\underline{r}_i) R_{nl}(r_i) y_{lm}(\theta_i, \phi_i). \tag{4.26}$$

Similar expressions may be written for the other potential types. The overall interaction energy may then be estimated as

$$E_{GRID} = \frac{1}{2}\int \left[\tau_A(\underline{r})\left(\phi_B^{\mathrm{N}+}(\underline{r}) + \phi_B^{\mathrm{O}-}(\underline{r}) + \phi_B^{\mathrm{Dry}}(\underline{r})\right) + \tau_B(\underline{r})\left(\phi_A^{\mathrm{N}+}(\underline{r}) + \phi_A^{\mathrm{O}-}(\underline{r}) + \phi_A^{\mathrm{Dry}}(\underline{r})\right)\right]\mathrm{d}V. \tag{4.27}$$

It should be noted that this expression does not specifically favour complementary pairings of individual probe types, nor does it penalise unfavourable pairings. However, by construction, it should give a deep minimum when the proteins are contraposed in a tightly fitting orientation. Hence this energy term should boost the score for complementary shape-based docking orientations. Figure 4.17 shows the calculated SPF N+ potential on the SAS of the SSI protein, along with the corresponding O- potential on SUP.

Compared to the shape-only and shape plus electrostatic docking calculations described above, we found that adding the above grid potential to the shape correlation function improved the rank of the first near-native solution by at least a factor of 2 (specifically, the best shape-only solution was found at rank 13, shape plus electrostatics gave rank 10, and shape plus grid gave rank 5). Unfortunately, because it required several manual steps to perform the PCA analyses and to import the data from GRID into *Hex*, it was not possible to perform more extensive tests during the Alessandra's time in Aberdeen. However, this work demonstrated for the first time the utility of correlating chemical potentials in docking calculations. The overall approach was subsequently used to produce some useful docking models of the CCR5 cell surface co-receptor protein and the MIP-1β and RANTES chemokine proteins (Fano *et al.*, 2006).

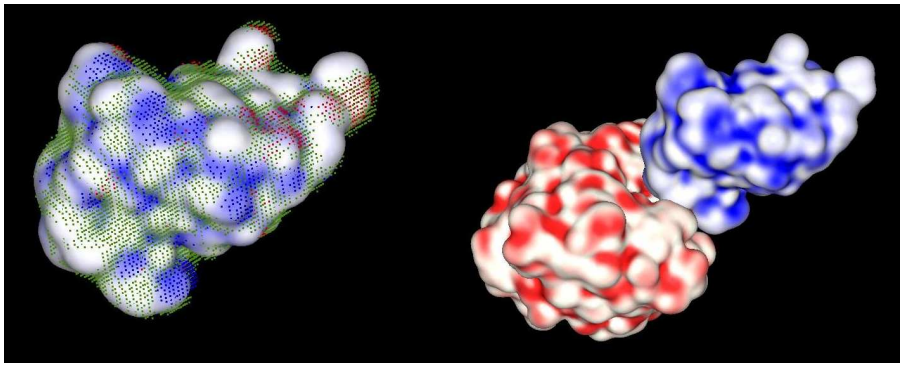

Figure 4.17: Left: the N+ potential (blue regions) calculated on the SAS (shown in white) of SSI. Blue dots show the positions of the original N+ GRID points, red dots show the O- positions and green dots show hydrophobic points. Right: the N+ (blue) and O- (red) probe hot-spots on the SSI and SUP SAS surfaces, respectively. This image shows the binding orientation of the complex but with the two proteins slightly separated for a better view.

4.2.6 Multi-Dimensional FFT Protein-Protein Docking

Because the FFT allows a problem that formally requires $O(N^2)$ operations to be computed in $O(N \log N)$ steps, it is reasonable to expect that greater computational speed-ups should be achieved when the FFT is applied to as many degrees of freedom as possible. This section describes how the SPF approach may be used to develop three-dimensional (3D) and five-dimensional (5D) docking correlation expressions. In order to achieve this, it is convenient to use both real and complex SH functions, where the two types of function are related by the unitary transformation matrix, $U^{(l)}$ (see Section 2.1.11),

$$y_{lm}(\theta,\phi) = \sum_{m'} U^{(l)}_{mm'} Y_{lm'}(\theta,\phi). \tag{4.28}$$

If a particular 3D property is initially sampled as a real expansion, it may be expressed in complex form using

$$A(\underline{r}) = \sum_{nlm}^{N} A_{nlm} R_{nl}(r) Y_{lm}(\theta,\phi) \tag{4.29}$$

where the complex expansion coefficients, A_{nlm}, are related to the original real coefficients by

$$A_{nlm} = \sum_{m'} U^{(l)}_{m'm} a_{nlm'}. \tag{4.30}$$

The correlation between a pair of complex properties, $A(\underline{r})$ and $B(\underline{r})$, may then be written as

$$E = \int (\hat{T}(-R)\hat{R}(0,\beta_A,\gamma_A)A(\underline{r}))^* (\hat{R}(\alpha_B,\beta_B,\gamma_B)B(\underline{r}))\mathrm{d}\underline{r}, \tag{4.31}$$

where the asterisk denotes complex conjugation. Substituting the SPF expansions then gives the complex version of Eq 4.1

$$E = \sum_{kjsmnlv} D^{(j)*}_{ms}(0,\beta_A,\gamma_A) A^*_{kjs} T^{(|m|)}_{kj,nl}(R) D^{(l)}_{mv}(\alpha_B,\beta_B,\gamma_B) B_{nlv}. \tag{4.32}$$

Now, summing over the k and n radial subscripts gives

$$E = \sum_{jsmlv} D^{(j)*}_{ms}(0,\beta_A,\gamma_A) S^{(|m|)}_{js,lv}(R) D^{(l)}_{mv}(\alpha_B,\beta_B,\gamma_B), \tag{4.33}$$

where the $S^{(|m|)}_{js,lv}(R)$ are the matrix elements of a reduced translation/overlap matrix given by

$$S^{(|m|)}_{js,lv}(R) = \sum_{kn} A^*_{kjs} T^{(|m|)}_{kj,nl}(R) B_{nlv}. \tag{4.34}$$

Eq 4.33 may be cast in exponential form by noting that a rotation of β about the y axis may always be calculated as a rotation of β about a rotated z axis using (Edmonds, 1957)

$$\hat{R}_y(\beta) \equiv \hat{R}_z(-\pi/2)\hat{R}_y(-\pi/2)\hat{R}_z(\beta)\hat{R}_y(\pi/2)\hat{R}_z(\pi/2), \tag{4.35}$$

and by re-writing the $d^l_{mm'}(\beta)$ matrices of a general Wigner rotation

$$D^{(l)}_{mm'}(\alpha,\beta,\gamma) = e^{-im\alpha} d^l_{mm'}(\beta) e^{-im'\gamma}, \tag{4.36}$$

as the corresponding product of complex exponentials

$$d^l_{mm'}(\beta) = \sum_t e^{im\pi/2} d^l_{mt}(-\pi/2) e^{-it\beta} d^l_{tm'}(\pi/2) e^{-im'\pi/2}. \tag{4.37}$$

Then, writing

$$\begin{aligned} \Delta^l_{tm} &= d^l_{tm}(\pi/2) \\ &= d^l_{mt}(-\pi/2), \end{aligned} \tag{4.38}$$

and collecting constant coefficients

$$\begin{aligned} \Gamma^{tm}_{lm'} &= e^{i(m-m')\pi/2} \Delta^l_{tm} \Delta^l_{tm'} \\ &= i^{m-m'} \Delta^l_{tm} \Delta^l_{tm'} \end{aligned} \tag{4.39}$$

allows the Wigner rotation matrix elements to be written in a completely exponential form

$$D^{(l)}_{mm'}(\alpha,\beta,\gamma) = \sum_t \Gamma^{tm}_{lm'} e^{-im\alpha} e^{-it\beta} e^{-im'\gamma}. \tag{4.40}$$

Substituting Eq 4.40 twice into Eq 4.33 then gives the fully factorised result

$$E = \sum_{jsmlvrt} \Gamma^{rm}_{js} S^{(|m|)}_{js,lv}(R) \Gamma^{tm}_{lv} e^{-i(r\beta_A - s\gamma_A + m\alpha_B + t\beta_B + v\gamma_B)}, \tag{4.41}$$

where the summation ranges over all subscript values that satisfy $|r| \leq j, |s| \leq j, |t| \leq l, |v| \leq l$, and $|m| \leq \min(l, j) \leq L$. In this equation, r and t enumerate azimuthal frequency components, and s, v, and m enumerate circular frequencies. I call Eq 4.41 the docking correlation master equation.

Equation 4.41 clearly has the form of a complex five-dimensional (5D) Fourier series. Hence it may be calculated using a multi-dimensional FFT. However, because Euler rotation angles have the ranges $0 \leq \alpha, \gamma < 360^{\circ}$ and $0 \leq \beta < 180^{\circ}$, it is useful to change the sign of the γ_A rotation and to scale the β rotation angles so that all rotational coordinates map to the natural phase and period of the FFT. If this is not done, the FFT calculation will over-sample the β coordinates to give duplicate solutions, each at half the desired resolution. Scaling the β coordinates eliminates this effect and allows a smaller FFT grid to be used, thus halving the amount of computer memory required for each β dimension and speeding up the FFT calculation.

Dealing with the sign of γ_A is straightforward. For example, putting $\gamma'_A = -\gamma_A$, and writing

$$e^{is\gamma_A} = \sum_q \eta_{sq} e^{-iq\gamma'_A}, \tag{4.42}$$

and using the orthogonality of the exponentials to solve for the coefficients, η_{sq}, gives

$$\eta_{sq} = \delta_{s\bar{q}}. \tag{4.43}$$

Similarly, the β rotations may be scaled by putting $\beta' = 2\beta$ and writing

$$e^{-it\beta} = \sum_u \lambda_{tu} e^{-iu\beta'}, \tag{4.44}$$

and again using the orthogonality of the exponentials to solve for the coefficients λ_{tu}. In this case, it can be shown using basic trigonometric relations that the coefficients are given by

$$\lambda_{tu} = \begin{cases} 2i/\pi(2u-t) & \text{if } t \text{ is odd,} \\ 1 & \text{if } t = 2u, \\ 0 & \text{otherwise.} \end{cases} \tag{4.45}$$

In other words, there exist exact solutions when t is even, and convergent power series solutions when t is odd. However, for current purposes, the coefficients λ_{tu} may be determined to reproduce *exactly* a finite set of M_β rotational samples by treating Eq 4.44 as a discrete Fourier transform analysis equation

$$\lambda_{tu} = \frac{1}{M_\beta} \sum_{n=0}^{M_\beta - 1} e^{-\pi i t n / M_\beta} e^{2\pi i u n / M_\beta}. \tag{4.46}$$

Other angular ranges (e.g. when performing a focused docking search) may be scaled onto the natural FFT period in a similar manner. Substituting the above changes of variable into Eq 4.41 and applying an inverse Fourier transform to the result gives

$$E[p,q,m,u,v;R] = \sum_{rt} \sum_{jl} \Gamma^{rm}_{j\bar{q}} S^{(|m|)}_{j\bar{q},lv}(R) \Gamma^{tm}_{lv} \lambda_{rp} \lambda_{tu}. \tag{4.47}$$

Collecting coefficients as

$$\Lambda^{um}_{lv} = \sum_t \Gamma^{tm}_{lv} \lambda_{tu} \tag{4.48}$$

finally gives

$$E[p,q,m,u,v;R] = \sum_{jl} \Lambda^{pm}_{j\bar{q}} S^{(|m|)}_{j\bar{q},lv}(R) \Lambda^{um}_{lv}. \tag{4.49}$$

This equation may be considered as an analytic recipe for calculating the array elements of a 5D FFT grid. Applying a forward Fourier transform to the elements of this array will produce a 5D array of $E(\beta_A, \gamma_A, \alpha_B, \beta_B, \gamma_B, R)$ function values for *unique* combinations of Euler rotation angles. Hence Eq 4.49 may be interpreted as an analytic generating function for 5D FFT docking correlations.

4.2.7 Multi-Dimensional FFTs

Although it is gratifying to be able to obtain a very compact formula for generating 5D FFT correlations, it does not automatically follow that a high order GF such as Eq 4.49 will be the most efficient to calculate in practice. For example, in Eq 4.49 it can be seen that the double sum over the jl subscripts means that the cost of initialising each 5D FFT grid cell scales as $O(N^2)$ and therefore the overall cost of setting up a 5D FFT scales as $O(N^7)$. Therefore, it is expedient to calculating Eq 4.49 as

$$W_{lv}^{pqm}(R) = \sum_{j} \Lambda_{j\bar{q}}^{pm} S_{j\bar{q},lv}^{(|m|)}(R) \tag{4.50}$$

and

$$E[p,q,m,u,v;R] = \sum_{l} W_{lv}^{pqm}(R)\Lambda_{lv}^{um}. \tag{4.51}$$

Thus, by using a temporary array, W, the $O(N^7)$ "set-up cost" of a 5D FFT can be computed practically using two $O(N^6)$ steps. The double sum in the expression for the reduced overlap matrix, Eq 4.34, may be calculated efficiently in a similar way. Nonetheless, using a large intermediate array makes significant additional demands on the available computer memory. One way to reduce the memory requirement is to set $\gamma_A = 0$ in the correlation expression and to explicitly rotate the receptor expansion coefficients to give

$$S_{jq,lv}^{(|m|)}(R,\gamma_A) = \sum_{kn}^{N} A_{kjq}^{*}(\gamma_A) T_{kj,nl}^{(|m|)}(R) B_{nlv} \tag{4.52}$$

where $A_{kjq}(\gamma_A)$ represents a rotated expansion coefficient. The 4D FFT array elements may then be calculated from the 4D GF

$$E[p,m,u,v;R,\gamma_A] = \sum_{jql} \Lambda_{j\bar{q}}^{pm} S_{j\bar{q},lv}^{(|m|)}(R,\gamma_A)\Lambda_{lv}^{um}. \tag{4.53}$$

Thus, in principle, a 6D docking search could be performed by iterating over pairs of (R,γ_A) samples and for each pair by calculating a 4D FFT over the remaining rotation angles. However, this approach can immediately be seen to be impractical because the triple sum in Eq 4.53 indicates that the set-up cost of initialising a 4D FFT grid is still $O(N^7)$. On the other hand, the GF complexity falls significantly if the β_A rotation angle is dropped from the FFT. For example, by explicitly transforming the real receptor expansion coefficients using Eq.s 4.3 and 4.4, and then calculating

$$A_{nlm}(R,\beta_A,\gamma_A) = \sum_{m'} U_{m'm}^{(l)} a_{nlm'}(R,\beta_A,\gamma_A), \tag{4.54}$$

the 3D GF is found to be

$$E[m,u,v;R,\beta_A,\gamma_A] = \sum_{l} S_{lv}^{m}(R,\beta_A,\gamma_A)\Lambda_{lv}^{um} \tag{4.55}$$

where

$$S_{lv}^{m}(R,\beta_A,\gamma_A) = \sum_{n} A_{nlm}^{*}(R,\beta_A,\gamma_A)B_{nlv}. \tag{4.56}$$

Hence (assuming pre-calculated receptor coefficient vectors, as before) it can be seen that the set-up cost for a 3D rotational FFT essentially scales as $O(N^2)$ per FFT grid cell. For the sake of completeness, the 2D GF has the same structure and set-up complexity as above, and may be stated as

$$E[m,u;R,\beta_A,\gamma_A,\gamma_B] = \sum_{lv} S_{lv}^{m}(R,\beta_A,\gamma_A,\gamma_B)\Lambda_{lv}^{um} \tag{4.57}$$

where

$$S_{lv}^{m}(R,\beta_A,\gamma_A,\gamma_B) = \sum_{n} A_{nlm}^{*}(R,\beta_A,\gamma_A)B_{nlv}(\gamma_B). \tag{4.58}$$

Therefore, like the 4D case, 2D correlations may be dismissed as being computationally impractical. The 1D GF (FFT set-up complexity $O(N^3)$ per α_B twist angle search) was implemented previously in real form (see Sections 4.2 and 4.1.1) and is given by

$$E[m;R,\beta_A,\gamma_A,\beta_B,\gamma_B] = \sum_{nl} A_{nlm}^{*}(R,\beta_A,\gamma_A)B_{nlm}(\beta_B,\gamma_B). \tag{4.59}$$

4.2.8 Multi-Property FFTs

It is well known that the correlation between two pairs of real properties may be calculated simultaneously using one complex FFT. For example, if the *in vacuo* electrostatic potential and charge density of a system of two proteins, A and B, are written as

$$\begin{aligned}\phi(\underline{r}) &= \phi_A(\underline{r}) + \phi_B(\underline{r}) \\ \rho(\underline{r}) &= \rho_A(\underline{r}) + \rho_B(\underline{r}),\end{aligned} \tag{4.60}$$

and if linear combinations of the SPF expansions are formed as

$$\begin{aligned}\underline{A} &= \underline{U}^T(\underline{a}^{\phi} + i\underline{a}^{\rho}) \\ \underline{B} &= \underline{U}^T(\underline{b}^{\rho} + i\underline{b}^{\phi}),\end{aligned} \tag{4.61}$$

where $\underline{U}^T$ is the transpose of the complex-to-real unitary transformation matrix $\underline{U}$ (c.f. Equations 4.28, and 4.30), then the electrostatic interaction energy for a pairwise orientation may be calculated as

$$E = Re(\underline{A}^{*}\underline{B}). \tag{4.62}$$

Similarly, dropping summation subscripts and using matrix notation for the 6D electrostatic interaction energy GF (Eq 4.49) gives

$$E[p,q,m,u,v;R] = \underline{\Lambda}^{pqm}\underline{S}^{qmv}(R)\underline{\Lambda}^{uvm}. \tag{4.63}$$

However, from the linearity of this expression, it follows that multiple interaction energy correlations $e = 0, 1, 2, ...$ may be computed simultaneously by first summing the distance-dependent part of each potential/density interaction

$$\left(\underline{S}_e^{qmv}(R)\right)_{jl} = \sum_{kn} A_{kjq}^{e*} T_{kj,nl}^{(|m|)}(R) B_{nlv}^{e}, \tag{4.64}$$

to give

$$E[p,q,m,u,v;R] = \underline{\Lambda}^{pqm}\Big(\sum_e \underline{S}_e^{qmv}(R)\Big)\underline{\Lambda}^{uvm}. \tag{4.65}$$

Thus, arbitrary combinations of correlations, *including* those which use different radial functions, may be evaluated together in a single 5D FFT with very little additional cost.

4.2.9 Multi-Resolution FFTs

It is worth noting that there is no requirement for the FFT grid dimensions to correspond exactly to the polynomial order of the SPF basis functions. For example, a low order GF may be evaluated on a high order FFT grid and *vice-versa*. This corresponds to padding the FFT grid with zeros or excluding frequency components that exceed the grid boundaries, respectively. Therefore, it is important to consider carefully both the polynomial expansion order and the FFT grid dimensions, because each can significantly influence overall performance. It was shown previously (Ritchie & Kemp, 2000; Ritchie, 2003) that the use of polynomial expansion orders in the range *L=24* to 30 is often sufficient to give satisfactory resolution when docking globular protein domains. According to Shannon sampling theory, this implies that an angular FFT grid dimension of at least *M=2L=48* should be used for thorough rotational sampling. This corresponds to using an angular search increment of $360^{\text{o}}/48 = 7.5^{\text{o}}$, which is somewhat finer than the rotational step sizes conventionally used in Cartesian FFT algorithms. Nonetheless, because two of the five rotational degrees of freedom can be described using Euler angles which range from 0 to 180°, it is evident that a 5D FFT grid of e.g. $48^3 \times 24^2$ cells can be accommodated in less than one gigabyte (Gb) of computer memory if grid values are stored as single precision floating point complex numbers (8 bytes per grid cell).

4.2.10 FFT Performance Comparison

As a first test of the utility of the multi-dimensional FFT approach, the HyHel-5/lysozyme complex (Figure 4.13) was docked at a range of expansion orders, L, using the conformation of the bound

antibody Fv fragment and unbound lysozyme. Table 4.3 presents a comparison of the accuracy and execution times of shape-only and shape plus electrostatic correlations for this example. All calculations sampled 53 translational steps of $\pm$0.75Å from the initial orientation of the complex. To facilitate comparison of the 3D and 5D correlations with the existing 1D radix-2 FFT implemented in *Hex*, $M_\alpha = 64$ was used for the twist angle dimension. The 3D and 5D grids each used $M_\gamma = 48$ and $M_\beta = 24$ to give (β, γ) increments of 7.5^o. The remaining rotational degrees of freedom in the 3D and 1D cases respectively used one and two icosahedral tessellations of the sphere, each of 812 vertices, to generate rotational samples with an average angular separation of around 7.7^o. Considering that the Euler grids tend to over-sample near the poles, this scheme gives broadly equivalent sampling densities with around 1.7, 2.5, and 3.5 billion docking orientations for the 1D, 3D, and 5D cases, respectively.

Table 4.3: Comparison of shape-only and shape plus electrostatic docking correlation times.

	1D Shape-Only		1D Shape+Electro		3D Shape-Only		3D Shape+Electro		5D Shape-Only		5D Shape+Electro	
L	Rank (RMS)	Time/m	Rank (RMS)	Time/m	Rank (RMS)	Time/m	Rank (RMS)	Time/m	Rank (RMS)	Time/m	Rank (RMS)	Time/m
16	646 (6.8)	28.7	428 (8.0)	52.0	864 (7.1)	15.1	254 (8.2)	18.1	–	37.5	669 (6.0)	40.3
20	336 (1.2)	52.7	20 (1.3)	102.7	410 (1.2)	23.5	17 (1.3)	29.2	336 (7.9)	39.3	29 (1.3)	46.5
24	417 (1.2)	92.4	52 (1.2)	184.2	501 (1.2)	33.2	53 (1.2)	51.2	833 (1.2)	53.0	82 (1.2)	56.2
26	49 (1.2)	123.3	15 (1.2)	243.1	48 (1.2)	43.5	15 (1.6)	69.0	45 (1.2)	58.7	13 (1.6)	63.1
28	54 (1.5)	158.1	8 (1.2)	315.6	22 (5.2)	54.2	11 (1.3)	92.2	19 (5.5)	64.5	13 (1.2)	71.7
30	113 (2.2)	203.5	43 (1.3)	403.0	47 (1.6)	69.8	20 (1.6)	122.5	61 (1.6)	74.3	19 (1.6)	108.0

In this table, L is the polynomial order of the expansion, Rank is the rank of the first orientation found in which the ligand is within 10Å RMS (shown in parentheses) of the crystal structure after clustering with the default *Hex* clustering threshold. A hyphen indicates no near-native orientation found within the top 2000 solutions. Time is the total computation time in minutes on a single processor 1.8GHz Pentium Xeon PC. The 3D and 5D FFT calculations used the Kiss FFT library.[2] For those calculations, the time spent within the FFT library is essentially constant at 13.1 and 34.3 minutes, respectively. All timings exclude the calculation of the translation matrix elements.

As expected, Table 4.3 shows that high order expansions generally assign a better rank to near-native orientations than low order expansions, but this trend is not necessarily monotonic. The best combination of a good rank and low ligand root mean squared deviation (RMSD) from the complex is typically obtained with *L=28* or *L=30*. This table also shows that shape-only 3D FFTs are around three times faster than the 1D calculation and, surprisingly, are also generally faster than 5D FFTs. However, due to the linearity of the GF, the cost of including electrostatics in 3D and 5D correlations is low compared to the cost of computing 1D shape plus electrostatic FFTs. Indeed, 5D FFTs of shape plus electrostatics are faster than 3D FFTs when *L*$\geq$*26*. These differences would become more pronounced if more potentials were included in the calculation.

4.2.11 Protein Docking Benchmark Results

The above approach was applied to the 84 complexes of the Protein Docking Benchmark (Mintseris *et al.*, 2005) using shape-only and shape plus electrostatic correlations (Ritchie *et al.*, 2008). As suggested by Table 4.3, a two-stage search protocol using 3D shape-only rotational FFT scans with *L=20* followed by 1D shape plus electrostatic re-scoring with *L=30* was used to obtain a good trade-off between speed and accuracy.

To provide a consistent pseudo-random starting orientation, all proteins were initially oriented by least-squares fitting to the complex, and a small off-grid rotation, $\hat{R}(\alpha,\beta,\gamma)$=$\hat{R}(11°,9°,0)$, was then applied to the ligand. The orientations calculated in each docking run were clustered using a greedy algorithm with a 9Å clustering threshold (Kozakov *et al.*, 2005), and the lowest energy member of each cluster was selected as the "solution" for that cluster. All other members of each cluster were discarded.

Seven different docking runs were performed for each complex to assess the shape-based and electrostatic components of the scoring function, and to investigate the difference between blind docking and the use of prior knowledge of one or both protein's binding sites. The results are shown in Table 4.4. The first set of figures in this table give the results for blind shape-only docking of bound subunits, presented as the rank and deviations of the first solution found within 10 Å RMS deviation of the complex (here called a "hit") along with the total number of such hits found within the top 2000 solutions. This threshold broadly corresponds to the definition of an "acceptable" prediction under the CAPRI assessment criteria (Méndez *et al.*, 2003). Although the final goal is to dock unbound subunits, consideration of bound docking results provides a practical way to identify complexes which will *a priori* be expected to be difficult to dock acceptably in the unbound case. Encouragingly, acceptable solutions are found within the top 10 in 33 cases, and within the top 20 in 37 cases. This shows that the *Hex* shape-based scoring function can often identify near-native crystallographic orientations.

However, these results also show that *Hex* fails to find an acceptable bound-bound solution for 22 of the Benchmark complexes. Visual inspection of these complexes shows that several (1AK4, 1GHQ, 1KTZ, 1BJ1, 1QFW, 2QFW, and 1ATN) have particularly small interface areas, which would therefore be expected to be difficult for any shape-based docking algorithm to identify. Furthermore, several of the other failing complexes include at least one large protein domain (e.g. 1KLU, 1ML0, 1KKL, 1HE8, 1N2C, 1DE4, 1H1V, and 2HMI) which cannot accurately be encoded in the standard *Hex* radial function. Hence, these cases will also be difficult for the *Hex* scoring function. Of the remaining failing complexes, several are antibody/antigen complexes (e.g. 1DQJ, 1E6J, 1WEJ, 2VIS), and it is generally not necessary to perform completely blind docking calculations on such well understood systems.

The rest of Table 4.4 presents results for docking unbound structures. As expected, the rank of the best shape-only blind docking solution is often considerably poorer compared to docking bound

Table 4.4: *Hex* Results for the Docking Benchmark (version 2).

	B-B Shape-Only Blind Search		U-U Shape-Only Blind Search		U-U Shape+Elec Blind Search		U-U Shape-Only One Constraint		U-U Shape+Elec One Constraint		U-U Shape-Only Two Constraints		U-U Shape+Elec Two Constraints	
Code	Rank (RMS)	Hits	Rank (RMS)	Hits	Rank (RMS)	Hits	Rank (RMS)	Hits	Rank (RMS)	Hits	Rank (RMS)	Hits	Rank (RMS)	Hits
Rigid-Body (63)														
1AVX	46 (4.8)	20	108 (8.9)	7	111 (8.9)	4	40 (8.9)	12	75 (9.0)	14	18 (9.0)	43	12 (9.0)	45
1AY7	40 (8.9)	16	645 (9.9)	4	–	–	99 (3.5)	20	234 (9.8)	1	17 (6.7)	39	17 (9.7)	18
1BVN	1 (1.1)	29	63 (9.1)	20	389 (9.6)	7	29 (9.6)	35	3 (6.6)	36	4 (5.1)	49	2 (9.6)	39
1CGI	1 (0.7)	24	42 (9.4)	17	47 (4.6)	9	20 (9.4)	14	42 (9.8)	11	4 (9.4)	31	4 (4.6)	24
1D6R	273 (1.3)	24	447 (7.7)	1	119 (7.6)	4	49 (7.7)	8	31 (7.7)	8	8 (7.7)	37	5 (7.7)	31
1DFJ	167 (4.2)	14	17 (9.5)	14	1 (4.2)	30	3 (9.5)	24	1 (4.2)	30	2 (9.5)	32	1 (4.2)	35
1E6E	1 (2.1)	14	109 (5.6)	10	5 (2.2)	24	24 (5.6)	19	3 (1.5)	29	5 (5.6)	38	1 (7.7)	49
1EAW	1 (1.0)	17	9 (5.0)	20	1 (4.0)	37	7 (5.0)	25	1 (4.0)	35	1 (5.0)	42	1 (4.0)	42
1EWY	19 (7.7)	16	76 (9.1)	12	24 (9.7)	14	114 (8.1)	12	103 (6.8)	7	9 (8.1)	37	9 (7.6)	23
1EZU	2 (0.9)	13	–	–	–	–	–	–	–	–	86 (6.7)	10	287 (6.2)	4
1F34	1 (1.4)	25	124 (6.7)	11	–	–	48 (7.1)	15	–	–	11 (5.4)	22	26 (6.5)	11
1HIA	3 (1.2)	30	51 (8.7)	6	8 (8.9)	15	72 (8.7)	21	15 (9.9)	22	15 (6.7)	33	6 (8.3)	32
1MAH	1 (0.9)	16	2 (1.2)	20	1 (1.1)	28	1 (1.2)	27	1 (1.2)	30	1 (1.2)	33	1 (1.2)	30
1PPE	1 (1.0)	42	2 (9.7)	47	4 (3.0)	31	1 (9.7)	49	1 (3.0)	46	1 (3.0)	43	1 (3.0)	45
1TMQ	1 (2.1)	19	356 (5.9)	9	427 (6.0)	6	45 (5.9)	21	264 (2.3)	7	7 (5.9)	39	10 (6.6)	38
1UDI	1 (1.6)	17	8 (6.2)	9	20 (6.2)	10	4 (6.2)	22	7 (6.2)	25	1 (6.2)	32	5 (6.2)	37
2MTA	11 (1.4)	18	136 (9.0)	4	79 (9.8)	20	38 (9.0)	17	12 (8.4)	24	15 (7.7)	33	15 (8.7)	31
2PCC	1007 (9.1)	1	–	–	18 (6.9)	33	14 (9.3)	20	12 (5.1)	31	5 (9.3)	37	14 (6.3)	44
2SIC	3 (0.7)	10	57 (8.8)	8	–	–	21 (8.9)	10	44 (1.0)	9	4 (8.9)	31	4 (1.0)	35
2SNI	1 (1.5)	18	256 (9.6)	7	101 (9.6)	6	39 (7.1)	15	40 (4.4)	11	5 (7.1)	31	5 (4.4)	25
7CEI	5 (1.3)	17	61 (8.7)	5	4 (8.4)	19	11 (8.7)	17	3 (8.4)	22	2 (8.7)	29	1 (8.4)	35
1AHW	6 (1.9)	10	234 (8.0)	3	7 (8.0)	12	31 (8.0)	12	5 (8.0)	40	3 (8.0)	42	5 (8.0)	38
1BVK	44 (1.5)	6	–	–	508 (6.7)	7	134 (9.4)	7	184 (6.8)	10	71 (9.9)	23	22 (6.8)	24
1DQJ	–	–	–	–	–	–	216 (8.6)	6	440 (9.9)	2	22 (8.6)	24	73 (8.1)	11
1E6J	–	–	–	–	–	–	26 (8.9)	12	16 (8.4)	22	2 (8.9)	37	4 (8.4)	41
1JPS	24 (1.3)	5	–	–	36 (8.8)	11	170 (6.6)	9	14 (6.6)	27	15 (6.6)	29	1 (8.8)	30
1MLC	62 (1.2)	5	408 (3.6)	2	–	–	25 (3.6)	13	22 (3.7)	28	3 (3.6)	29	2 (3.7)	23
1VFB	23 (1.1)	3	–	–	–	–	97 (9.1)	14	51 (7.1)	10	14 (9.1)	36	12 (7.1)	35
1WEJ	–	–	–	–	–	–	26 (1.7)	13	2 (1.7)	20	8 (1.7)	29	1 (1.7)	37
2VIS	–	–	–	–	–	–	–	–	–	–	–	–	–	–
1A2K	29 (5.4)	12	–	–	–	–	–	–	–	–	186 (9.3)	5	274 (9.1)	4
1AK4	–	–	–	–	–	–	–	–	–	–	–	–	–	–
1AKJ	30 (8.4)	25	209 (9.6)	10	17 (9.4)	27	110 (6.3)	15	23 (2.7)	35	23 (9.6)	36	5 (9.6)	48
1B6C	3 (1.8)	19	593 (9.0)	2	755 (8.9)	2	88 (9.0)	5	133 (8.5)	5	19 (9.0)	27	7 (9.7)	36
1BUH	28 (1.0)	9	743 (7.7)	2	289 (7.8)	4	52 (7.7)	14	19 (7.7)	13	28 (7.7)	19	8 (7.7)	18
1E96	133 (1.1)	5	–	–	302 (8.6)	2	246 (9.4)	6	119 (8.6)	8	37 (9.7)	13	43 (8.5)	20
1F51	3 (1.4)	21	371 (9.6)	5	–	–	149 (9.6)	12	58 (9.3)	3	9 (7.6)	19	8 (7.5)	27
1FC2	605 (6.5)	2	–	–	–	–	–	–	–	–	–	–	297 (7.7)	10
1FQJ	7 (1.0)	14	41 (8.0)	12	7 (7.9)	14	14 (8.0)	21	7 (7.7)	28	5 (7.8)	31	4 (7.7)	41
1GCQ	1 (1.0)	16	–	–	–	–	–	–	–	–	92 (6.2)	6	–	–
1GHQ	–	–	–	–	–	–	828 (8.9)	2	–	–	30 (8.9)	13	175 (6.7)	6
1HE1	1 (1.5)	24	37 (6.4)	18	88 (6.3)	15	10 (6.4)	26	28 (7.2)	25	2 (7.6)	39	9 (7.2)	39
1I4D	31 (1.5)	19	–	–	–	–	–	–	–	–	505 (8.1)	1	481 (9.4)	1
1KAC	36 (1.2)	7	687 (8.7)	1	271 (8.9)	5	7 (4.4)	19	4 (4.4)	26	4 (4.4)	33	2 (4.4)	32
1KLU	–	–	–	–	–	–	–	–	–	–	591 (9.7)	2	–	–
1KTZ	–	–	–	–	–	–	–	–	–	–	238 (9.4)	4	25 (6.0)	10
1KXP	1 (1.1)	22	36 (9.4)	13	1 (7.5)	13	15 (9.4)	19	1 (6.9)	30	7 (9.4)	24	1 (6.9)	29
1ML0	–	–	–	–	–	–	7 (9.1)	8	33 (7.0)	11	1 (9.1)	22	3 (5.6)	27
1QA9	86 (5.9)	7	–	–	161 (9.9)	3	587 (7.5)	8	481 (6.8)	4	25 (5.3)	28	23 (4.5)	28
1RLB	409 (8.8)	2	–	–	–	–	–	–	–	–	305 (6.3)	7	384 (6.3)	6
1SBB	–	–	–	–	–	–	–	–	–	–	–	–	–	–
2BTF	5 (0.8)	8	–	–	–	–	133 (8.6)	13	16 (6.7)	22	32 (8.6)	19	4 (6.7)	34
1BJ1	–	–	–	–	–	–	–	–	–	–	7 (6.7)	13	10 (6.9)	10
1FSK	10 (1.3)	16	5 (1.8)	16	6 (1.4)	10	1 (1.8)	31	1 (1.8)	31	1 (1.8)	43	1 (1.8)	46

(continued)

Table 4.2: (continued).

Code	B-B Shape-Only Blind Search Rank (RMS)	Hits	U-U Shape-Only Blind Search Rank (RMS)	Hits	U-U Shape+Elec Blind Search Rank (RMS)	Hits	U-U Shape-Only One Constraint Rank (RMS)	Hits	U-U Shape+Elec One Constraint Rank (RMS)	Hits	U-U Shape-Only Two Constraints Rank (RMS)	Hits	U-U Shape+Elec Two Constraints Rank (RMS)	Hits
1I9R	5 (5.7)	14	82 (2.1)	8	4 (2.1)	15	23 (2.1)	19	13 (2.1)	26	7 (2.1)	29	5 (2.1)	26
1IQD	42 (0.7)	8	–	–	760 (1.4)	3	276 (6.1)	7	5 (6.1)	16	5 (9.4)	27	3 (6.1)	29
1K4C	24 (0.7)	4	21 (9.6)	1	–	–	4 (9.6)	3	311 (9.6)	2	2 (9.6)	17	46 (9.6)	19
1KXQ	6 (5.5)	10	488 (7.1)	5	35 (6.3)	12	48 (7.1)	16	27 (7.1)	15	27 (7.1)	18	24 (7.1)	16
1NCA	1 (1.1)	11	116 (1.2)	5	139 (1.9)	3	20 (1.2)	13	8 (0.9)	16	2 (9.9)	22	3 (0.9)	30
1NSN	11 (1.7)	8	142 (1.5)	6	–	–	18 (1.5)	19	14 (1.5)	12	6 (1.5)	22	3 (1.5)	23
1QFW	–	–	–	–	–	–	–	–	–	–	333 (6.3)	3	37 (6.3)	6
2QFW	–	–	–	–	–	–	–	–	–	–	522 (9.7)	1	–	–
2JEL	10 (1.1)	10	164 (6.0)	3	–	–	7 (6.0)	27	4 (5.6)	29	6 (6.0)	39	2 (6.0)	38
Mean	**25 (4.1)**	**11**	**242 (8.4)**	**5**	**156 (8.1)**	**7**	**66 (7.6)**	**13**	**46 (7.0)**	**14**	**15 (7.3)**	**25**	**13 (6.7)**	**25**
Medium Difficulty (13)														
1ACB	36 (0.9)	8	694 (8.3)	3	674 (8.5)	2	156 (8.3)	7	163 (8.3)	1	10 (8.3)	33	88 (8.4)	14
1KKL	–	–	–	–	–	–	48 (8.6)	18	94 (8.4)	10	8 (8.7)	40	14 (8.0)	31
1BGX	1 (3.0)	3	–	–	–	–	–	–	–	–	–	–	–	–
1GP2	–	–	–	–	419 (6.9)	5	–	–	137 (7.1)	8	113 (5.6)	12	68 (7.1)	17
1GRN	1 (1.3)	13	914 (9.1)	2	586 (2.5)	5	661 (7.1)	4	27 (6.3)	23	14 (7.4)	31	20 (6.3)	29
1HE8	–	–	–	–	–	–	–	–	–	–	–	–	–	–
1I2M	1 (1.8)	17	–	–	29 (5.4)	24	754 (8.5)	3	15 (8.5)	24	107 (6.7)	14	21 (8.5)	24
1IB1	10 (5.0)	13	–	–	–	–	–	–	–	–	14 (9.8)	13	22 (9.9)	7
1IJK	189 (3.0)	10	1012 (8.7)	3	–	–	145 (8.7)	5	383 (8.7)	1	14 (8.7)	18	70 (8.7)	5
1K5D	406 (5.9)	4	–	–	146 (7.6)	3	–	–	128 (9.1)	5	377 (7.6)	4	21 (9.7)	17
1M10	429 (9.1)	4	514 (9.5)	2	48 (9.2)	4	130 (9.5)	4	46 (9.3)	6	13 (9.5)	8	124 (8.4)	12
1N2C	–	–	–	–	–	–	–	–	–	–	–	–	–	–
1WQ1	1 (1.5)	26	125 (7.1)	10	16 (7.2)	17	34 (7.1)	14	13 (7.1)	20	6 (7.1)	27	3 (7.1)	33
Mean	**50 (5.5)**	**8**	**782 (9.5)**	**1**	**329 (8.2)**	**5**	**306 (8.8)**	**5**	**153 (8.7)**	**8**	**58 (8.4)**	**15**	**66 (8.6)**	**15**
Difficult (8)														
1ATN	–	–	–	–	–	–	–	–	–	–	–	–	–	–
1DE4	–	–	946 (8.6)	1	15 (8.4)	3	164 (8.6)	3	–	–	184 (8.5)	8	35 (9.9)	8
1EER	1 (4.0)	25	609 (9.2)	8	43 (9.2)	16	106 (7.6)	18	30 (7.7)	18	34 (7.6)	23	39 (7.7)	13
1FAK	–	–	–	–	–	–	–	–	–	–	768 (7.0)	2	221 (7.0)	8
1FQ1	162 (5.6)	5	–	–	–	–	469 (8.4)	2	–	–	82 (8.4)	5	508 (8.4)	3
1H1V	–	–	–	–	–	–	–	–	–	–	–	–	–	–
1IBR	4 (3.0)	27	–	–	–	–	–	–	–	–	314 (8.8)	4	68 (8.4)	6
2HMI	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Mean	**168 (7.8)**	**7**	**933 (9.7)**	**1**	**399 (9.7)**	**2**	**549 (9.3)**	**3**	**359 (9.3)**	**3**	**325 (8.8)**	**5**	**238 (8.9)**	**5**

In this table, B-B and U-U denote bound-bound and unbound-unbound docking, respectively. A hyphen denotes no acceptable solution within the top 2000, in which case a value of 10Å is used when calculating the mean RMS deviation. Means of ranks were calculated using the MLR formula, Eq 4.66. For the antibody/antigen complexes (1AHW, 1BVK, 1DQJ, 1E6J, 1DQJ, 1JPS, 1MLC, 1VFB, 1WEJ, 2VIS, 1BJ1, 1FSK, 1I9R, 1IQD, 1K4C, 1KXQ, 1NCA, 1NSN, 1QFW, 2QFW, 2JEL, 1BGX, 2HMI), the Cα coordinates of heavy chain residue 37 were used as the antibody coordinate origin. For all other structures, the centre of mass was used as the coordinate origin. It should be noted that the Docking Benchmark includes several antibody complexes (1BJ1, 1FSK, 1I9R, 1IQD, 1K4C, 1KXQ, 1NCA, 1NSN, 1QFW, 2QFW, 2JEL, 2HMI) for which only the *bound* antibody Fab coordinates are available.

components, with only 6 complexes being ranked within the top 20. On the other hand, including the ETO electrostatic interaction term in the correlation often improves the rank of the best solution, giving 16 complexes within the top 20. However, using electrostatic correlations can worsen the prediction in some cases, but it is not clear how to predict which those cases might be *ab initio*.

Nonetheless, in practice, it is becoming increasingly rare that completely blind docking is necessary because, like the antibody families, biochemical or biophysical knowledge is often available to indicate the identities of key interaction residues. Hence, four further constrained docking runs were performed for each complex to simulate such data-driven docking scenarios. Here, the range of the FFT searches were constrained by applying the restriction $\beta_A \leq 45^\circ$ to simulate using knowledge of the receptor binding site (tabulated as "One Constraint"), and additionally $\beta_B \leq 45^\circ$ corresponding to using knowledge of both the receptor and ligand binding sites ("Two Constraints"). These constraints each reduce the size of the search space and corresponding FFT grid dimensions by a factor of about four, and speed up the FFT scan correspondingly. Thus, for constrained docking runs, overall calculation times of just a few minutes arise largely from the *L=30* re-scoring stage. Specifying a receptor constraint of $\beta_A = 45^\circ$ would physically correspond to spinning an antigen over the antibody hypervariable loop region in an antibody/antigen complex, as illustrated in Figure 4.1, for example. In general, *Hex* allows a given receptor and ligand residue to be rotated onto the z axis before each docking run. Hence, for example, by setting small values for the β_A and β_B angular ranges, it is straightforward to focus a docking calculation around a given pair of residues in a known or hypothesized protein-protein interface.

As can be seen from Table 4.4, the above rather loose constraints are often sufficient to improve considerably the rank of near-native solutions. For example, using only the receptor constraint is sufficient to increase the rate of acceptable solutions from 6 to 17 within the top 20. Adding the *Hex* electrostatic correlation term boosts this improvement to 28 within the top 20. Applying a similar ligand constraint further improves the success rate to 48 in the top 20 and 35 in the top 10 for shape only correlations, or 45 in the top 20 and 37 in the top 10 for shape plus electrostatics. In other words, the electrostatic component helps significantly to identify the general orientation of the binding mode, and it can also help to distinguish a near-native orientation from amongst high ranking shape-based orientations, although the improvement in the latter is less dramatic. It is worth noting that constrained docking also improves the results for several complexes that rigid-body docking indicated would be difficult (specifically 1GHQ, 1KTZ, 1ML0, 1BJ1, 1QFW, 1KKL, and 1DE4).

In order to compare such trends more objectively, Table 4.4 presents overall average results for each set of calculations. Here, we calculate the mean rank using the mean of the logarithm of the rank (MLR) of each first acceptable hit according to:

$$\mathrm{MLR} = \exp\{\frac{1}{N_C}\sum_{i=1}^{N_C}\ln(\min(\mathrm{Rank}_i, 1000))\}, \tag{4.66}$$

where N_C is the number of complexes in each Benchmark category. Limiting poor results to a value of 1000 in this formula helps to prevent outliers from adversely biasing the overall score. Hence the MLR score ranges from 1 (rank 1 hits for all complexes) to 1000 (no hits for any complex). The MLR figures in Table 4.4 readily show the benefit of using just one, or preferably two, loose constraints to enrich the number of high ranking predictions in each Benchmark category. This benefit is most dramatic

in the Rigid-Body category, although using two constraints also significantly enhances the results for both the Medium Difficulty and Difficult categories.

Overall, blind 3D shape-only docking correlations find acceptable solutions within the top 20 in 6 cases, whereas including electrostatics in the calculation gives 16 solutions within the top 20. Applying a single loose angular constraint to focus the calculation around the receptor binding site is sufficient to produce acceptable solutions within the top 20 in 28 cases. Further constraining the search to the ligand binding site in a similar manner gives up to 48 solutions within the top 20.

In terms of raw processing speeds, 3D shape-only and shape plus electrostatic FFTs are found to be around three times faster than the 1D FFT initially implemented in *Hex* (see Section 4.1.1) but, surprisingly, 3D FFTs are also often faster than 5D FFTs. On the other hand, thanks to the linearity of the FFT, multiple properties may be correlated simultaneously in the 5D FFT, and this is expected to be particularly advantageous when calculating high order correlations of multi-term knowledge-based protein-protein interaction potentials.

4.2.12 Simulating Protein Flexibility During Docking

It should be emphasized that rigid-body docking is normally only the first stage of a docking calculation. If a protein complex consists of N atoms, there are *3N-6* internal degrees of freedom available to the constituent atoms. Because proteins often consist of a few hundred amino acid residues, or equivalently several thousand atoms, the dimensionality of this space is truly vast. However, the internal atomic coordinates are not completely independent but are in fact highly constrained by the covalent bonds between the atoms. Nonetheless, using MD to simulate the atomic fluctuations and internal motions within large proteins is computationally very expensive. On the other hand, it is known from both experimental and MD simulation evidence that often only a relatively small number of bond torsion angles change significantly when two proteins form a complex. The unsolved challenge, however, is to identify and incorporate into the calculation only those flexible regions of a protein that need to be modelled during docking.

Currently, the most promising approach to reduce the dimensionality of the flexible docking problem seems to be the use of so-called “slow-mode” or normal mode analysis (NMA) techniques (Hinsen *et al.*, 1999). These approaches are derived from the use of matrix diagonalisation techniques to analyse the large-scale atomic fluctuations within a protein during MD simulations. It is often observed that only a small number of the most significant eigenvectors, or principal components, of the fluctuation matrix can account for the bulk of the motion within a protein. In other words, there are only a few (typically no more than around 20) of mutually independent degrees of freedom (Amadei *et al.*, 1993). More recently, techniques have been developed to compute rapidly and approximately the principal components whilst avoiding the computational expense of performing a full MD simulation (Tirion, 1996; de Groot *et al.*, 1997).

As part of her PhD project at Aberdeen to simulate protein flexibility during protein-protein docking simulations, Diana Mustard investigated the use of an eigenvector analysis based approach to sample protein conformational space and hence to generate multiple feasible protein conformations for rigid-body docking in *Hex*. The overall idea is that protein flexibility would be simulated by docking multiple rigid-body conformations generated from the initial protein structures. This approach was applied to nine of the protein complex targets (targets T8–T14, T18, and T19) of the CAPRI blind docking experiment (Mustard & Ritchie, 2005), and was presented at the Second CAPRI Evaluation Meeting (2004) in Gaeta, Italy.

In order to generate a large number of initial 3D protein conformations, we used the distance-constraint based CONCOORD and DISCO programs (de Groot *et al.*, 1997) to generate an ensemble of pseudo-random 3D structures. This ensemble of structures may be considered as permissible sample points within the multi-dimensional conformational space of the protein. Figure 4.18 shows some of the conformations of the Laminin protein (CAPRI target T8) generated by CONCOORD. To capture the most significant fluctuations within this space, the essential dynamics (ED) approach constructs a square covariance matrix $\underline{C}$ of the means of the deviations of each atom's coordinates x_i from its initial unbound position u_i:

$$C_{ij} = < (x_i - u_i)(x_j - u_j) >, \tag{4.67}$$

where the subscripts $i, j = 1...3N$ label the components of the Cartesian coordinates of the N atoms under consideration, and where the angle-brackets denote taking the average over all pseudo-random structures sampled. Hence a least $3N$ conformational samples are required for an ED analysis. Because the covariance matrix is square-symmetric, it may be factorised according to:

$$\underline{C} = \underline{T}.\underline{\Lambda}.\underline{T}^T \tag{4.68}$$

where $\underline{T}$ is the matrix of eigenvectors, $\underline{e_k}$, and $\underline{\Lambda}$ is a diagonal matrix of eigenvalues, λ_k. The eigenvectors and eigenvalues represent the principal components and squared magnitudes of the coordinate fluctuations in $\underline{C}$, respectively. If the eigenvectors are considered in order of decreasing size of their corresponding eigenvalues, most of the fluctuation is found within the first few eigenvectors. Hence the distance constraint essential dynamics (DCED) technique captures much of the internal motion of a protein whilst avoiding most of the computational expense of running a full MD simulation (Amadei *et al.*, 1993). Figure 4.19 shows that the first few eigenvectors capture most of the internal motion within the Laminin protein.

Here, we adapted the above approach to generate feasible conformations for rigid-body docking in *Hex* by first performing a DCED analysis on the initial conformation of the starting structure. Because the eigenvectors are orthonormal and span the conformational coordinate space of the protein, any 3D conformation may, in principle, be constructed from an appropriate combination of eigenvectors. For example, denoting vectors of bound and unbound protein coordinates by $\underline{B}$ and $\underline{U}$ (with $\underline{U} =$

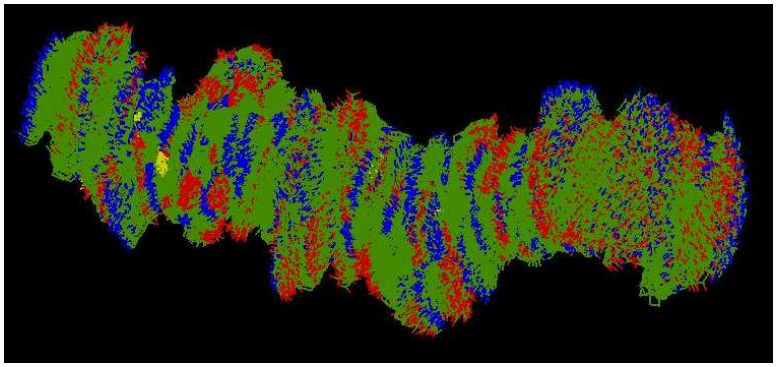

Figure 4.18: Multiple rigid-body conformations of the Laminin protein (CAPRI target T8) generated by CONCOORD.

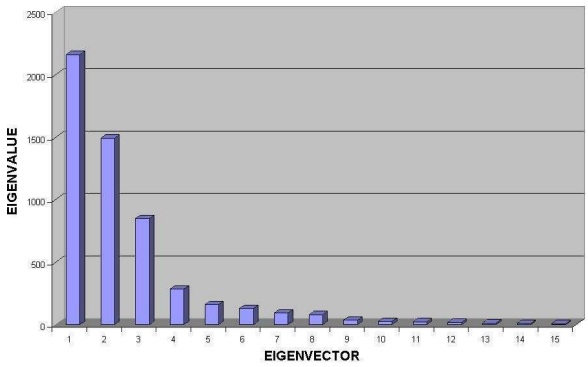

Figure 4.19: The magnitudes of the largest eigenvectors for the Laminin protein (CAPRI target T8). This figure shows that first few eigenvectors describe most of the internal motion.

$\{u_i; i = 1...3N\}$, etc.), we may write

$$\underline{B} = \underline{U} + \sum_k \alpha_k \underline{e}_k. \tag{4.69}$$

The coefficients α_k represent the weights with which the eigenvectors should be combined in order to obtain the bound conformation from the coordinates of the unbound structure. When the unbound and bound coordinates are known, we call the quantity

$$\underline{V} = \underline{B} - \underline{U} \tag{4.70}$$

the "docking vector." If the coordinates of both the bound and unbound structures are available, the weights α_k may be solved exactly using a projection:

$$\alpha_k = \underline{V}.\underline{e}_k. \tag{4.71}$$

This provides a useful way to evaluate the approach using the structures of known protein complexes. Figure 4.20 shows the C_α RMS deviation between the calculated and actual bound conformations as a function of the number of eigenvectors used in Eq 4.69. This shows that the first few eigenvectors can account for much of the backbone conformational change that takes place on binding. Of course, in predictive docking the desired bound conformation is not available, and indeed the initial unbound conformation may have been model-built. Nonetheless, it is reasonable to assume that following an ED analysis of the starting structure, there will exist some combination of the most significant eigenvectors which will transform the initial structure into one that resembles more the bound form. In other words, we assume that each conformation is accessible from the other through the fluctuation modes embedded in its structure. The task then reduces to calculating the weights with which a sufficient number of eigenvectors should be combined.

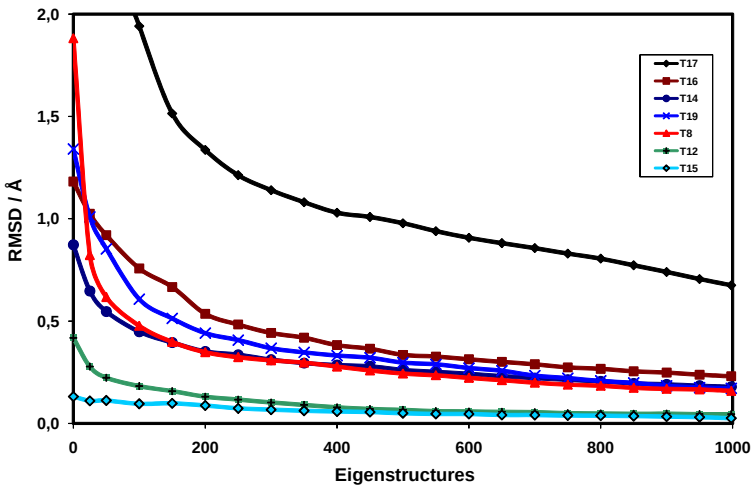

Figure 4.20: RMS deviation between the calculated and actual bound conformations of the CAPRI ligand structures as a function of the number of eigenvectors used in Eq 4.69. Eigenvectors are combined using weights α_k calculated from a projection (Eq 4.71) of the unbound-bound docking vector (Eq 4.70). The plots show that most of the conformational change that occurs on binding may be accounted for by the first few eigenvectors.

In our approach, the DCED analysis is applied to the heavy atoms C_α, C, O, N, and C_β (if present). Then, each candidate backbone conformation $\underline{B}_{nj}$ is constructed as:

$$\underline{B}_{nj} = \underline{U} + \sum_{k=1}^{n} \alpha_{kj} \underline{e}_k \tag{4.72}$$

where the subscript j enumerates samples along the kth eigenvector, $\alpha_{kj} = \pm\delta, \pm 2\delta$, etc. We use $\delta = 0.25$Å. Thus each candidate conformation deviates from all others by an integral multiple of 0.25Å RMS. However, many of these conformations will have infeasible covalent bond lengths and angles.

Hence, we arbitrarily define any covalent bond which differs by more than 1% from the original structure as a "bad bond," and we reject any conformation with more than five bad bonds. Figure 4.21 shows the distribution of bad bonds for the Laminin protein eigenstructures. In our current implementation, up to n=8 eigenvectors are sampled subject to the constraint that $|\alpha_{kj}| \leq \sqrt{\lambda_k}$, and this can produce up to around 10^5 candidate backbone conformations. Applying our simple bond length filter typically reduces this number to less than 100. Because these candidate structures have locally very similar backbone geometries to the starting conformation, side chain atom coordinates are transferred directly from the starting structure. We call the resulting 3D protein structures "eigenstructures." In the current study, eigenstructures were generated for just one of the proteins in each complex (typically the unbound or smaller component) to avoid the computational expense of multiple cross-dockings.

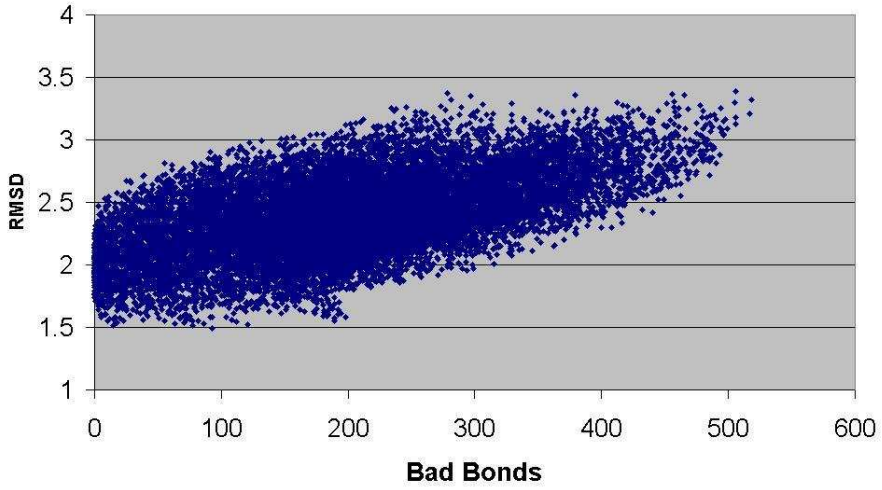

Figure 4.21: Bond length violations in the first 8 eigenvectors for the Laminin protein.

Table 4.3 summarises the number of conformational models generated and filtered for the CAPRI targets using the above procedure. This table omits the very flexible T9 LiCT structure, and T18 is also omitted because the TAXI ligand in that target does not have a contiguous backbone, as required by CONCOORD. The final three columns of RMS deviations show that in all cases conformations can be generated which have a lower C_α deviation from the bound form than the initial unbound structure. For example, when using a projection on the first eight eigenvectors, the unbound-bound C_α deviation for T8 is reduced by 0.58Å RMS from 1.88 to 1.30Å RMS. For T17, the corresponding reduction is 0.70Å RMS. With blind eigenvector sampling using a fixed step size, the reduction available in the RMS deviations is less than the theoretical optimum, but it can still be significant in favourable cases (e.g. 0.26 and 0.21Å RMS, for T8 and T17, respectively). Hence, Table 4.3 shows that improved

backbone conformations may be generated with relatively little effort in the DCED approach.

Table 4.3: Summary of the eigenstructures generated for the targets in CAPRI rounds 3–5.

Target	Ligand	AA	EV	CS	ES	B/UB	B/ES(opt)	B/ES(δ)
T8	Laminin	162	8	405,405	624	1.88	1.30	1.62
T10	TBEV-B	395	6	8,505	49	11.33	10.90	11.13
T12	Dockerin	138	5	1,215	19	0.44	0.37	0.42
T13	SAG1	129	8	54,675	52	0.96	0.92	0.94
T14	PP1	294	8	59,535	229	0.88	0.83	0.85
T15	ImmD	87	5	2,025	6	0.15	0.11	0.15
T16	GH10	257	8	120,285	49	1.19	1.12	1.16
T17	GH11	188	8	54,675	33	5.09	4.39	4.88
T19	PrP	121	7	28,431	54	1.59	1.26	1.47

This table summarises the number of conformational models generated and filtered for targets T8–T19 of CAPRI rounds 3–5. The table columns are labelled as follows: AA: the number of amino acids in the ligand structure; EV: the number of eigenvectors used; CS: the number of candidate structures generated from the eigenvectors using a step size of $\delta = 0.275$Å; ES: the number of eigenstructures remaining after applying the bond length filter; B/UB: the C_α RMS deviation between the bound and unbound structures; B/ES(opt): the best possible RMS deviation between the bound conformation and the optimal eigenstructure calculated from projections of the first 8 eigenvectors; B/ES(δ): the lowest C_α RMS deviation between the bound conformation and the best eigenstructure found when using a fixed step size (δ) search along the first 8 eigenvectors.

In order to investigate the utility of our eigenstructure docking approach, we retrospectively docked ED-generated eigenstructures for several of the CAPRI targets which were amenable to analysis by the DCED approach. Because the crystal structures of the complexes had been revealed, we chose to start each docking run with the eigenstructures initially superposed onto the C_α atoms of the complex. All docking runs in this test used shape-only *N=30* correlations and 45 degree angular search constraints on each protein. In some cases, the eigenvector step size and bond length filter parameters were modified slightly (e.g. for T8 and T14) in order to achieve a more manageable list of eigenstructures (< 100) to be docked. Each list of eigenstructures was then seeded with the conformations of the unbound and bound ligand structures in order to facilitate comparison of docking the three types of structure. Due to the large number of orientational samples generated, all docking solutions were sorted and clustered as described previously (Ritchie, 2003). The total computational cost was around 12 CPU-hours per complex on a 1.8GHz AMD Athlon processor.

Table 4.4: Eigenstructure docking results for the CAPRI rounds 3–5 targets.

Target	Docked ES	Bound	RMS	Unbound	RMS	ES	RMS
T8	94	84(1/2)	9.71	30(40/94)	8.80	30(1/94)	8.24
T11	37	19(1/5)	5.52	2(29/183)	9.55	2(1/183)	9.20
T12	40	1(1/90)	0.64	1(23/90)	1.53	1(6/90)	1.53
T13	52	5(1/9)	1.17	1(32/306)	0.96	1(1/306)	6.24
T14	60	16(1/3)	9.95	10(10/177)	8.81	10(1/177)	8.81
T15	39	20(6/17)	7.80	8(20/77)	3.47	8(1/77)	4.94
T17	33	3(1/8)	1.56	–	–	12(1/43)	8.64
T19	40	1(1/12)	0.95	13(46/66)	7.70	13(1/66)	5.28

This table lists the number of ligand eigenstructures docked for each target followed by the cluster rank of the first solution with a C_α deviation of 10Å RMS or less obtained when docking bound, unbound, and DCED-generated eigenstructures, respectively. Figures in brackets (n/m) give the rank of the orientation (n) within the cluster of given size (m). A hyphen denotes no low RMS solution found within the first 512 clusters.

Overall, our initial studies on DCED eigenstructures seem very promising. We have shown that the first few eigenvectors intrinsically encode much of the backbone conformational flexibility observed on binding. The results with the CAPRI targets show that docking multiple eigenstructures generated from the first eight eigenvectors is sufficient to give better docking predictions than docking only the initial unbound or model-built structures. However, there is clearly scope to improve the quality of the generated conformations. For example, increasing the number of eigenvectors sampled and using a variable step size to search along each eigenvector should give better coverage of the conformational space accessible to each protein. Additionally, using a more sensitive method of estimating and possibly minimising the internal energies of the generated eigenstructures should give more physically realistic structures within that space. However, it will also be necessary to make the docking scoring function more selective in order to identify a near-native conformation of the complex from a large repertoire of physically realistic decoys.

4.3 Small-Molecule Virtual Screening

Broadly speaking, there are two main approaches to virtual screening. In receptor-based approaches, the structure of the protein target is known or has been modelled, and the goal is to find suitable ligands which will bind near the receptor active site and consequently antagonise (block) the native receptor function, for example. In ligand-based approaches, the structure of the protein target is generally not known, and the goal is to find new ligands which are similar to known antagonists. Some drug molecules act as agonists (i.e. they activate or enhance the native receptor function), but the screening principles are essentially the same as for antagonists. Due to the computational expense of receptor-based (i.e. docking-based) approaches, high throughput virtual screening (HTVS)

campaigns often employ a combination of approaches, in which ligand-based similarity criteria are used as an initial filter, and candidate compounds which survive this filter are subsequently docked to the protein target.

Currently, the most commonly used techniques for rapidly searching large chemical databases in ligand-based virtual screening employ bit-string representations of molecular properties and topologies such as Daylight, UNITY, and MACCS fingerprints. However, by their nature, these representations have a tendency to find close chemical analogues to the given query, which may not be sufficiently novel to be worth pursuing in a drug development programme. On the other hand, the pharmacological action of most drug molecules is governed by their interaction with their biological targets *via* ligand-receptor binding. Therefore, molecules that have similar overall shapes might reasonably be expected to bind to a protein in similar ways. However, comparing 3D molecular shapes is significantly more computationally expensive than comparing bit-strings (Lemmen & Lengauer, 2000).

In my opinion, the current state of the art for efficient 3D molecular shape comparison is based on Gaussian representations of molecular shape (Grant *et al.*, 1996) and the more recent SH surface envelope approach, which was developed independently by myself at Aberdeen (Ritchie & Kemp, 1999; Mavridis *et al.*, 2007), Tim Clark at Erlangen (Lin & Clark, 2005), and Bernard Maigret at Nancy (Cai *et al.*, 2002; Yamagishi *et al.*, 2006). At Aberdeen, the focus was to exploit the rotational properties of the SH functions to develop a very fast way of superposing and quantitatively comparing the 3D shapes of molecular surfaces. At Erlangen, the aim was to represent key QM molecular surfaces properties using the SH representation. At Nancy, the main objective was to use SH representations to provide a fast shape-based filter for protein-ligand docking. This approach has recently been incorporated into the VSM-G project (Beautrait *et al.*, 2008). To complement the Erlangen work, I recently developed the ParaFit program which can superpose 3D molecular structures calculated by ParaSurf at a rate of up to one hundred molecules per second on a single processor. ParaFit and ParaSurf are currently being marketed by Cepos Insilico Ltd.

4.3.1 The ParaFit Program

ParaFit superposes and compares molecules using SH expansions of the molecular surface and local surface properties calculated by ParaSurf (Lin & Clark, 2005). By exploiting the special rotational properties of the spherical harmonic basis functions (Wigner, 1939; Rose, 1957), computation times can be reduced by several orders of magnitude compared to conventional shape matching algorithms (Ritchie & Kemp, 1999; Ritchie & Kemp, 2000). ParaFit provides three main calculation modes. In the default “fitting” mode, ParaFit superposes one or more moving molecules onto a single fixed reference molecule. The program can also perform all-versus-all superpositions in which each molecule is superposed in turn onto all others. In this “matrix” mode, a table of distance scores is written out in a format suitable for subsequent clustering analysis, for example. In addition to superposing molecules,

ParaFit may also be used to align molecules to the coordinate axes in order to place them in a standard or “canonical” orientation. This is often a useful first step in QSAR studies. ParaFit can also apply arbitrary coordinate transformations to a given list of SDFs (Structure Description Files) created by ParaSurf. These transformations could be supplied as part of a processing pipeline by other superposition programs that do not have the capability to rotate complex QM properties such as quadrupole and octupole moments and atomic orbital charge density matrix elements. The ability of ParaFit to rotate all of the orientation-dependent QM information in an SDF eliminates the need to recalculate expensive QM quantities for new molecular orientations.

4.3.2 Spherical Harmonic Surface Shape Similarity

SH molecular surface shapes are represented as two-dimensional radial expansions of the form

$$r(\theta,\phi) = \sum_{l=0}^{L} \sum_{m=-l}^{l} a_{lm} y_{lm}(\theta,\phi), \tag{4.73}$$

where $y_{lm}(\theta,\phi)$ are normalized real spherical harmonic functions and a_{lm} are the expansion coefficients. The coordinates are defined with respect to the harmonic coordinate origin (CoH), which is usually equivalent to the molecular centre of gravity (CoG). In order to calculate a superposition between a pair of molecules, ParaFit translates the CoH of the moving molecule (B) to that of the fixed reference molecule (A) and then searches for the rotation that minimizes the “distance” between the corresponding pairs of spherical harmonic expansions:

$$D_{\text{Euclidean}} = \int_0^{\pi} \int_0^{2\pi} \left(r_A(\theta,\phi) - \hat{R}(\alpha,\beta,\gamma) r_B(\theta,\phi) \right)^2 \sin\theta \mathrm{d}\theta \mathrm{d}\phi. \tag{4.74}$$

By rotating the expansion coefficients using the real Wigner rotation matrices (Eq 3.35), and by exploiting the orthonormality of the basis functions, this expression reduces to

$$D_{\text{Euclidean}} = |\underline{a}|^2 + |\underline{b}|^2 - 2\underline{a}.\underline{b}'. \tag{4.75}$$

where $\underline{b}'$ represents the vector of rotated SH expansion coefficients of the moving molecule, etc. This function has units of Å^2 and clearly depends on the relative size of the molecules being compared. This is called a Euclidean distance function due to its analogy to Euclidean distances in ordinary 3D space. However, when comparing multiple molecules, it is often convenient to use normalized similarity functions in which identical molecules give a score of unity. For example, dividing by the sum of the magnitudes of the SH shape vectors gives the Hodgkin similarity score:

$$S_{\text{Hodgkin}} = \frac{2\underline{a}.\underline{b}'}{|\underline{a}|^2 + |\underline{b}|^2} = 1 - \frac{D_{\text{Euclidean}}}{|\underline{a}|^2 + |\underline{b}|^2}. \tag{4.76}$$

ParaFit also implements the Carbo and Tanimoto similarity functions:

$$S_{\text{CARBO}} = \frac{\underline{a}.\underline{b}'}{|\underline{a}|^2.|\underline{b}|^2}. \tag{4.77}$$

and

$$S_{\text{TANIMOTO}} = \frac{\underline{a}.\underline{b}'}{|\underline{a}|^2 + |\underline{b}|^2 - \underline{a}.\underline{b}'}. \tag{4.78}$$

ParaFit uses the Tanimoto function as its default similarity function. It is generally not obvious which of the above scoring functions is to be preferred. In my experience they all give good pairwise superpositions with the default *L=6* SH expansion order.

ParaFit superposes molecules using a brute-force rotational search over the three Euler rotation angles. Conceptually, each moving molecule is rotated with respect to the fixed reference molecule, and the Euler rotation that gives the greatest similarity (or smallest distance) score is recorded. This is essentially a Fourier correlation search in Euler angle coordinates. However, because good superpositions may be achieved using only low order harmonic expansions, it is not necessary to use fast Fourier transform (FFT) techniques to accelerate the calculation unless *L*$\geq$*16* (see Section 4.1.1).

In addition to using low order correlation searches, ParaFit's superposition calculations are accelerated in two further ways. The first technique exploits the fact that harmonic expansions to order L can have no more than L^2 local maxima. Hence, ParaFit initially uses relatively large angular search steps of around 8° to cover the search space. In order to sample angular space evenly and efficiently, these angular samples are generated from the vertices of an icosahedral tessellation of the sphere (Section 2.3). For a given angular step size, this gives around 30% fewer sample points than a naïve equi-angular grid (Ritchie & Kemp, 1999). Once the approximate location of maximum similarity has been identified, it is then refined using a localized grid search in steps of 2°. Both angular step sizes may be adjusted by the user.

The second acceleration technique is used when comparing multiple molecules. Rather than separately rotating each of the moving molecules in turn, it is much more efficient to rotate the SH expansions of only the reference molecule and to compare these against each of the moving molecules. Thus, relatively expensive SH rotations are applied to just one rather than N molecules. Once the optimal rotations have been found, the moving molecules are rotated using the inverse of the corresponding reference rotations. Using these techniques, a pair of molecules may be superposed in around 0.05 seconds on a 1.8GHz Pentium Xeon processor, and computation times may be further reduced by a factor of up to five when multiple molecules are compared in a single ParaFit run.

4.3.3 Rotation-Invariant Fingerprints and Canonical Orientations

Although pairs of molecules may be superposed and compared very quickly using the above techniques, it is necessary to develop even faster comparison techniques in order to search very large 3D structural databases (e.g. $> 10^6$ molecules) for HTVS. It is therefore natural to use the vector interpretation of SH coefficients to construct a rotationally invariant fingerprint (RIF) for each SH molecular shape. Noting that expansion coefficients with the same value of l transform amongst themselves

under rotation, the RIF coefficients are defined as:

$$A_l = \left(\sum_{m=-l}^{m=l} a_{lm}^2 \right)^{1/2} \tag{4.79}$$

and

$$A_L = \left(\sum_{l=0}^{L} A_l^2 \right)^{1/2}. \tag{4.80}$$

By analogy to Equation 4.75, the RIF distance score between a pair of molecules may be calculated as

$$D_{RIF} = A_L^2 + B_L^2 - 2 \sum_{l=0}^{L} A_l B_l. \tag{4.81}$$

A detailed study of the performance of the above rotation-dependent and rotation-invariant scoring functions has been published (Mavridis *et al.*, 2007). This study showed that very good superpositions may be obtained using only *L=3* expansions, and that using expansion orders above *L=6* gives little or no improvement in the quality of the results compared to high order *L=15* comparisons which were treated as the "gold standard".

Although RIF comparisons may be calculated very rapidly, they are significantly less accurate than the more expensive rotational correlation searches. Therefore, we investigated the possibility of comparing molecular shapes in standard "canonicalised" orientations as a way of retaining much of the precision of the rotational comparisons whilst avoiding the computational expense of a full rotational search. Here, a canonical orientation is calculated by rotating a SH surface shape expansion to *L=6* such that its largest radial extent is aligned with the global z axis, and by then applying a pure z axis rotation to place its maximal equatorial extent on the positive x axis. This procedure is somewhat similar to aligning the moments of inertia with the principal axes (Lanzavecchia *et al.*, 2001), but using expansions with *L*$>$*2* eliminate any ambiguity with respect to 180° axis flips. Figure 4.22 illustrates the canonical orientations of four benzodiazepine molecules.

4.3.4 Clustering and Classifying The Drug and Odour Datasets

In order to compare the SH surface comparison approach with other traditional molecular similarity measures, cluster analyses of two molecular datasets, here called "Drug" and "Odour," were performed and results were compared with conventional physico-chemical (PC) clustering of the Drug dataset and with the vibrational frequency based clustering results of the Odour dataset obtained previously by Takane and Mitchell (2004). This work was also carried out as part of the PhD thesis project of Lazaros Mavridis.

The Drug dataset was initially classified by my colleague Dr Brian Hudson into six broad pharmacological categories and up to seven sub-groups based on known pharmacological mechanisms

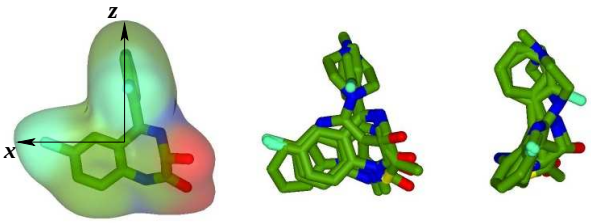

Figure 4.22: Illustration of the canonical alignment of four GABA receptor agonists, the benzodiazepines lLorazepam, diazepam, temazepam, and clonazepam. Left: The *L=6* SH molecular surface and canonicalised orientation of Lorazepam; Middle: The four canonicalised benzodiazepines together; Right: the same orientations rotated by 90° about the z axis.

of action. This gave a total of 22 drug classes, as shown in Table 4.5. It should be noted that this classification is not unique because many of these compounds have multiple modes of action and are used for a variety of therapeutic purposes. Nonetheless, the expert classification does provide an indication of pharmacological similarity against which calculated clusters may be compared.

Table 4.5: Pharmacological classification of the 73 molecules of the Drug dataset.

Name	Class	World Drug Index Keywords
MINOCYCLINE	AB 1	ANTIBIOTICS
DOXYCYCLINE	AB 1	ANTIBIOTICS
TETRACYCLINE	AB 1	ANTIBIOTICS
CEFPROZIL	AB 1	ANTIBIOTICS
CLINDAMYCIN	AB 1	ANTIBIOTICS
IBUPROFEN	AI 1	ANALGESICS; ANTIINFLAMMATORIES; ANTIPYRETICS
ASPIRIN	AI 1	ANALGESICS; ANTIINFLAMMATORIES; ANTIPYRETICS; ANTICOAGULANTS
DICLOFENAC	AI 1	ANALGESICS; ANTIINFLAMMATORIES; PROSTAGLANDIN-ANTAGONISTS
NAPROXEN	AI 1	ANALGESICS; ANTIINFLAMMATORIES; PROSTAGLANDIN-ANTAGONISTS; ANTIPYRETICS
CODEINE	AI 1	ANALGESICS; ANTITUSSIVES; NARCOTICS
CARISOPRODOL	AI 1	ANALGESICS; RELAXANTS
LORATADINE	AI 2	ANTIHISTAMINES-H1
CETIRIZINE	AI 2	ANTIHISTAMINES-H1
PROMETHAZINE	AI 2	ANTIHISTAMINES-H1; SEDATIVES
TRIAMCINOLONE	AI 3	CORTICOSTEROIDS
METHYLPREDNISOLONE	AI 3	CORTICOSTEROIDS
BUDESONIDE	AI 3	CORTICOSTEROIDS
PREDNISONE	AI 3	CORTICOSTEROIDS
CLONAZEPAM	CN 1	ANTICONVULSANTS
GABAPENTIN	CN 1	ANTICONVULSANTS
PHENYTOIN	CN 1	ANTICONVULSANTS
TOPIRAMATE	CN 1	ANTICONVULSANTS
SERTRALINE	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
FLUOXETINE	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
NORTRIPTYLINE	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
AMITRIPTYLINE	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
PAROXETINE	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
CITALOPRAM	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
BUPROPION	CN 2	ANTIDEPRESSANTS; PSYCHOSTIMULANTS
OLANZAPINE	CN 3	PSYCHOSEDATIVES; DOPAMINE-ANTAGONISTS; NEUROLEPTICS
RISPERIDONE	CN 3	PSYCHOSEDATIVES; NEUROLEPTICS; ANTISEROTONINS; DOPAMINE-ANTAGONISTS
LORAZEPAM	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS
BUSPIRONE	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS
DIAZEPAM	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS
TEMAZEPAM	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS; ANTICONVULSANTS
TRAZODONE	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS; PSYCHOSTIMULANTS; ANTIDEPRESSANTS
CYCLOBENZAPRINE	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS; RELAXANTS
ZOLPIDEM	CN 3	PSYCHOSEDATIVES; TRANQUILIZERS; SEDATIVES
FENOFIBRATE	CV 1	ANTIARTERIOSCLEROTICS
GEMFIBROZIL	CV 1	ANTIARTERIOSCLEROTICS
SIMVASTATIN	CV 1	ANTIARTERIOSCLEROTICS; HMG-COA-REDUCTASE-INHIBITORS
PRAVASTATIN	CV 1	ANTIARTERIOSCLEROTICS; HMG-COA-REDUCTASE-INHIBITORS
WARFARIN	CV 2	ANTICOAGULANTS
NIFEDIPINE	CV 3	CARDIANTS; CALCIUM-ANTAGONISTS
DILTIAZEM	CV 3	CARDIANTS; CALCIUM-ANTAGONISTS
VERAPAMIL	CV 3	CARDIANTS; CALCIUM-ANTAGONISTS; PROTEIN-KINASE-C-INHIBITORS
TRIAMTERENE	CV 4	DIURETICS
SPIRONOLACTONE	CV 4	DIURETICS; ALDOSTERONE-ANTAGONISTS
HYDROCHLOROTHIAZIDE	CV 4	DIURETICS; CARBONIC-ANHYDRASE-INHIBITORS; HYPOTENSIVES
FUROSEMIDE	CV 4	DIURETICS; PROTEIN-KINASE-C-INHIBITORS
VALSARTAN	CV 5	HYPOTENSIVES
TERAZOSIN	CV 5	HYPOTENSIVES
CAPTOPRIL	CV 5	HYPOTENSIVES; ANGIOTENSIN-ANTAGONISTS
FOSINOPRIL	CV 5	HYPOTENSIVES; ANGIOTENSIN-ANTAGONISTS
DOXAZOSIN	CV 5	HYPOTENSIVES; SYMPATHOLYTICS-ALPHA
BISOPROLOL	CV 5	HYPOTENSIVES; SYMPATHOLYTICS-BETA; ANTIARRHYTHMICS
CARVEDILOL	CV 5	HYPOTENSIVES; SYMPATHOLYTICS-BETA; VASODILATORS
CLONIDINE	CV 5	HYPOTENSIVES; SYMPATHOMIMETICS-ALPHA
ATENOLOL	CV 6	SYMPATHOLYTICS-BETA
METOPROLOL	CV 6	SYMPATHOLYTICS-BETA
TIMOLOL	CV 6	SYMPATHOLYTICS-BETA
FAMOTIDINE	GI 1	GASTRIC-SECRETION-INHIBITORS; ANTIHISTAMINES-H2
RANITIDINE	GI 1	GASTRIC-SECRETION-INHIBITORS; ANTIHISTAMINES-H2; ANTIULCERS
LANSOPRAZOLE	GI 2	GASTRIC-SECRETION-INHIBITORS; H-K-ATPASE-INHIBITORS
OMEPRAZOLE	GI 2	GASTRIC-SECRETION-INHIBITORS; H-K-ATPASE-INHIBITORS; ANTIULCERS
GLIPIZIDE	OT 1	ANTIDIABETICS
METFORMIN	OT 1	ANTIDIABETICS
METOCLOPRAMIDE	OT 2	ANTIEMETICS; DOPAMINE-ANTAGONISTS
ALLOPURINOL	OT 3	ANTIGOUTS; ANTIRHEUMATICS
CARBIDOPA	OT 4	ANTIPARKINSONIANS; DOPA-DECARBOXYLASE-INHIBITORS
ESTRADIOL	OT 5	ESTROGENS
FLUCONAZOLE	OT 6	FUNGICIDES
TAMOXIFEN	OT 7	PROTEIN-KINASE-C-INHIBITORS; ESTROGEN-ANTAGONISTS

Each drug molecule has been assigned a two-letter code by a chemoinformatics expert according to its main pharmacological class as follows, AB: antibiotic; AI: anti-inflammatory; CN: central nervous system; CV: cardio-vascular; GI: gastro-intestinal; OT: other. The numeric digits indicate sub-classes within each of the six main pharmacological classes.

For the PC clustering, 11 molecular descriptors (including polarizibility, radius of gyration, molecular weight, logP, etc.) were calculated for the Drug dataset using Cerius-2,[3] and these were auto-scaled and clustered using Ward's agglomerative clustering algorithm (Ward, 1963) to produce a total of 22 clusters, as shown in Figure 4.23. For the SH clustering, a shape-only distance matrix for the same group of molecules using *L=6* expansions was calculated and Ward's algorithm was applied directly to produce 22 clusters. These are also shown in Figure 4.23. This figure shows that both clustering methods often group similar classes of drug into the same or similar clusters. For example, both the PC and SH clustering approaches group the antibiotics (AB) together, and both approaches group many of the tranquilizer and anti-depressant drugs (central nervous system; CN) closely together. This would suggest that classifying molecules using SH surfaces is at least as good as traditional methods based on overall molecular properties. Indeed, comparison of the two dendrograms suggests that the SH clustering tends to place more of the pharmacologically related molecules into more closely related groups than the PC clustering. For example, the SH clustering places the gastro-intestinal (GI) drugs in the same group, whereas these drugs are distributed over four distinct groups in the PC clustering. For the CN compounds, one group contains the benzodiazepines clonazepam, lorazepam and diazepam, and a second group primarily contains compounds related to GPCR activity such as the serotonin re-uptake inhibitors amitryptiline, nortryptiline, citalopram, fluoxetine, and paroxetine as well as the serotonin receptor antagonist olanzapine. These features of the cluster analysis are not conclusive but are nevertheless encouraging.

As a further test of the SH approach, the 46 molecules of the Odour dataset was clustered into ten groups using SH shape descriptors to *L=6* using both rotational superpositions and by comparing the molecules in their pre-aligned canonical orientations. Takane and Mitchell (2004) originally clustered this dataset into ten distinct groups using eigenvalue (EVA) descriptors derived from quantum mechanical vibrational frequency calculations. Hence the same number of clusters was used in the present study to facilitate comparison with the SH results. Figure 4.25 shows the resulting SH clusters along with the corresponding 3D superpositions. Both the SH and EVA methods give broadly similar groupings. However, the SH clustering nicely distinguishes the camphoraceous and bitter almond molecules as two separate groups, whereas the earlier EVA clustering study splits the camphors into two sub-groups, one of which includes one jasmine and two rose odours (see Table 3 of Takane and Mitchell, 2004). The EVA clustering also splits the bitter almond odours into three distinct groups, whereas the SH clustering correctly assigns these molecules to two neighbouring sub-groups. The SH clustering also locates all but one of the rose and jasmine odours in two closely related sub-groups. Overall, Figure 4.25 shows a striking correspondence between the SH shape-based classification and the corresponding molecular shape superpositions.

The clustering results of both the Drug and Odour datasets show that SH shape comparisons often give chemically meaningful groupings. Our results for the Odour dataset show a striking corre-

[3] http://www.accelrys.com/.

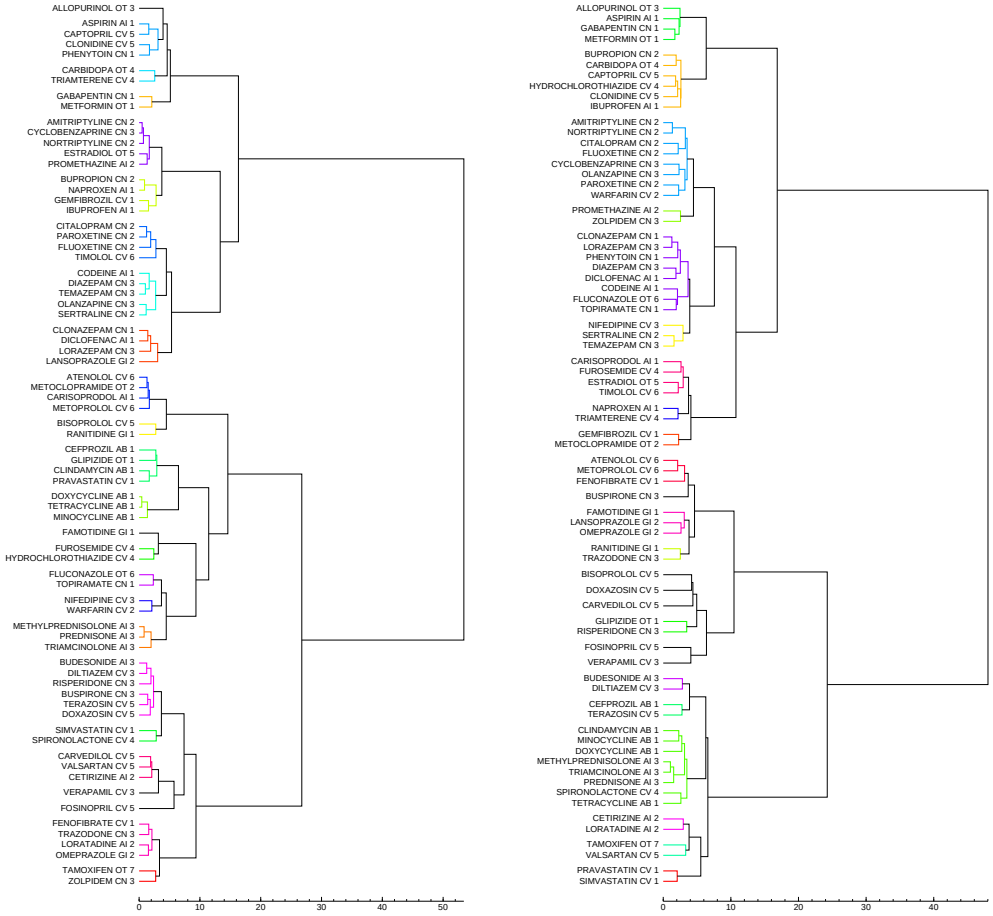

Figure 4.23: Dendrograms of the 73 molecules of the Drug dataset, calculated using Ward's agglomerative clustering algorithm to give 22 clusters. Left: conventional chemical clustering using 11 auto-scaled macroscopic PC descriptors. Right: *L=6* SH surface shape clustering.

spondence between the SH-based classification and the corresponding molecular shape superpositions. Indeed, SH shape clustering achieves comparable or better clusters than previous work based on more computationally expensive quantum mechanics-based vibrational frequency analysis. The analysis of the drug dataset is also encouraging in that, despite using only shape expansions, it is possible to identify similar pharmacological groupings which appear to be at least as good as those produced using traditional PC analyses.

Figure 4.26 shows the results of clustering the Odour dataset using SH surface comparisons of molecules in their canonical orientations. By comparing these results with Figure 4.25, it can be seen that comparing molecules in canonical orientations is almost as reliable as performing full pair-wise rotational comparisons. Furthermore, if the molecular SH coefficient vectors are compared in their canonicalised orientations, the computational cost of the comparison is essentially no more than that

of a rotationally invariant comparision.

Further experiments on the Drug dataset, which has been augmented with a large number of similar decoy molecules, have confirmed that canonical comparisons are much more accurate than rotational invariant comparisons (Mavridis *et al.*, 2007). It can therefore be concluded that for best database performance with the SH representation, molecular SH coefficients should be pre-calculated and stored in the database using canonicalised orientations.

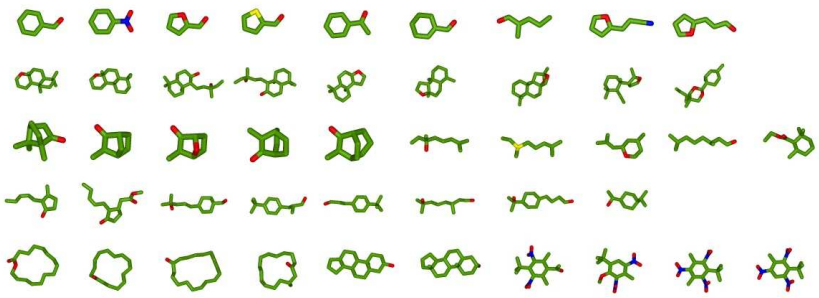

Figure 4.24: The 46 molecules in the Odour dataset of Takane and Mitchell (2004). These molecules may be classified into 7 principal odours: bitter, ambergris, camphoraceous, rose, jasmine, muguet, and musk.

4.3.5 Virtual Screening HIV Entry-Blockers

According to the World Health Organization, about 33 million people currently live with Acquired Immune Deficiency Syndrome (AIDS).[4] The principal cause of AIDS is infection by the human immunodeficiency virus (HIV), which at the molecular level begins with binding of the gp120 viral envelope glycoprotein to both the CD4 cell surface receptor and one of CXCR4 or CCR5 chemokine coreceptors which consequently leads to fusion of the viral capsid with the cell membrane. Current antiretroviral therapies (ARTs) against AIDS are generally based on reverse transcriptase inhibitors and protease inhibitors. Despite advances in the development of these agents which block HIV transcription and assembly, there remain problems regarding drug resistance, latent viral reservoirs, and drug induced toxic effects, which can all compromise effective control of the virus. Hence there is considerable interest in developing new classes of anti-HIV drugs with different modes of action. One very promising approach towards this goal has been the development of so-called entry-blocking drugs which interfere with the initial interaction between the viral gp120 protein and the host CXCR4 and CCR5 cell surface receptor proteins. Many small molecule CXCR4 and CCR5 antagonists have already been identified, and the first commercial HIV entry blocker, Maraviroc, was recently released. However, there remains an on-going need to develop new and more potent HIV entry blockers (Carrieri *et al.*,

[4] http://www.who.int/hiv/en/.

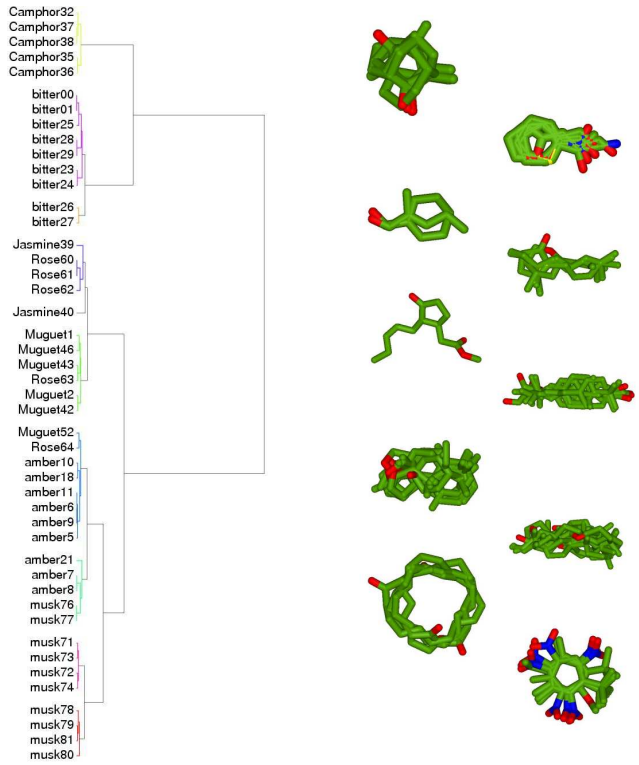

Figure 4.25: SH surface shape clustering of the Odour dataset. Left: the dendrogram obtained using *L=6* SH shape similarity calculations. Right: the corresponding 3D molecular superpositions.

2009).

As part of my contribution towards the above goal, over the last five years I have developed some very rewarding and productive collaborations with my computational chemistry colleagues Prof Antonio Carrieri at the University of Bari (Italy) and Prof Jordi Teixidó at the Institut Químic de Sarriá which is part of the Universitat Ramon Llul in Barcelona (Spain). Both of these collaborations involved extended visits to my group in Aberdeen of PhD students Alessandra Fano from Bari and Violeta Pérez-Nueno from Barcelona.

The SH shape matching approach implemented in the ParaFit and ParaSurf programs was thoroughly evaluated by Violeta Pérez-Nueno as part of her PhD project in VS of HIV entry inhibitors. In order to perform VS of ligands for the CXCR4 and CCR5 receptor proteins, Violeta first compiled a large dataset of 248 CXCR4 and 354 CCR5 known inhibitors from the literature, which mainly consist of 5 compound families for the CXCR4 inhibitors and 13 compound families for the CCR5 inhibitors. She also assembled a database of 4696 presumed inactive decoy compounds with similar physicochemical properties to the actives assembled from the Maybridge Screening Collection[5]. Retrospec-

[5]http://www.maybridge.com/

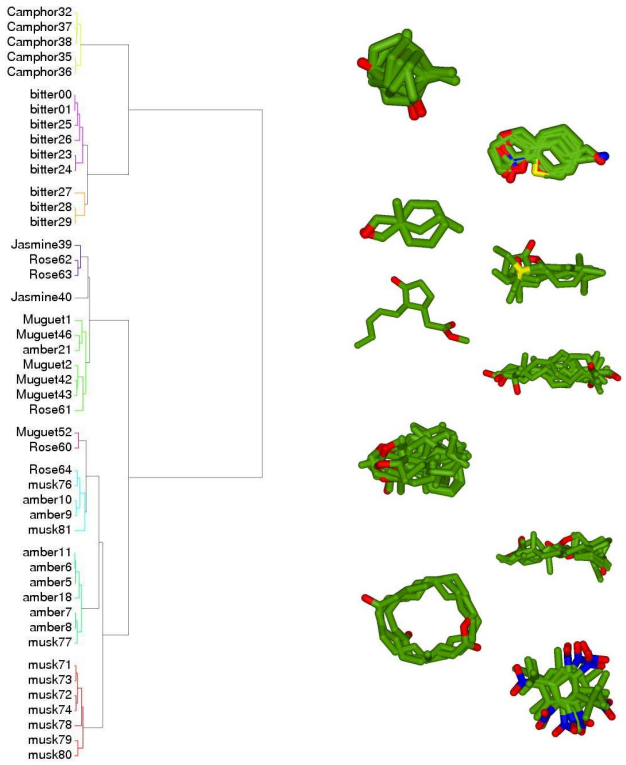

Figure 4.26: SH canonical surface shape clustering of the Odour dataset. Left: the dendrogram obtained using *L=6* with ten clusters; Right: the corresponding 3D molecular superpositions.

tive VS of the CXCR4 inhibitors was then performed by using AMD3100 (a known high-affinity ligand for CXCR4) as the “query” molecule, and by calculating the similarity between it and each of the 248 known CXCR4 ligands and 4696 decoys. Similarly, the CCR5 inhibitors were screened using the high affinity TAK779 molecule as the query which was compared against each of the 354 known CCR5 ligands and the 4696 decoys.

In computational chemistry, it is common practice to assess VS experiments using either enrichment factor (EF) plots or Receiver-Operator-Characteristic (ROC) plots. Both approaches are broadly equivalent to plotting recall against precision in conventional information retrieval analyses. However, ROC plots are becoming increasingly popular because they provide a more objective way to compare experiments that use different numbers of positive and negative instances. Nonetheless, EF plots were used initially in the present case because existing spreadsheet macros were available to perform the calculations. For both kinds of analysis, the compounds are sorted into a ranked list according to their similarity with the query, and the list is then analysed to assess how many of the actives appear near the top of the list. More specifically, in an EF analysis, at each position in the list the ratio of the number of known actives (or hits), $\text{Hits}_{\text{sampled}}$, relative to the number of compounds

sampled this far, $N_{sampled}$, is compared to the ratio of the total number of actives, $Hits_{total}$, relative to total number of molecules in the database, N_{total}. In other words, the EF is calculated at each position of the ranked list using:

$$EF = \frac{Hits_{sampled}/N_{sampled}}{Hits_{total}/N_{total}}. \tag{4.82}$$

Figure 4.27 shows the VS enrichment plots obtained when comparing the ParaFit 2D SH shape-matching function and the *Hex* 3D density matching function with the industry-standard ROCS shape-only and shape-plus-chemical "Combo" scoring functions. This figure shows that the ROCS Combo score gives the best EFs when using AMD3100 and TAK779 as queries. However, it can also be seen that the ParaFit 2D shape Tanimoto gives generally better results than ROCS shape Tanimoto and often gives comparable results to the ROCS Combo score for both CXCR4 and CCR5 inhibitors. The *Hex* 3D shape Tanimoto functions performs well for the CXCR4 inhibitors at first percentages of database screened, but the EFs are considerably lower for the CCR5 inhibitors. For the CXCR4 inhibitors, the *Hex* shape Tanimoto and ROCS Combo score give EFs comparable to the theoretical maximum (19.9%) at the first percentage of database screened. For the CCR5 inhibitors, the ROCS Combo score and the ParaFit Tanimoto score give EFs comparable to the theoretical maximum (14.3%) at the first percentage of database screened. Moreover, for the CXCR4 inhibitors, the four shape matching scoring functions perform well at the next percentages of database screened. However, the CCR5 inhibitor EFs are generally not as good as the CXCR4 EFs, although the relative utility of the different scoring functions is similar in both cases. The lower EFs obtained for CCR5 appear to be because the query conformation is not able to superpose well onto all of the CCR5 ligand families. The query superposes well onto actives from the same scaffold (which are retrieved first) but it cannot superpose well to actives with different scaffolds.

This study also compared the utility of ligand-based VS with receptor-based VS of both the CXCR4 and CCR5 proteins using a variety of protein-ligand docking tools. It is worth noting that, at least for these receptors, the ligand-based screening approaches showed better VS enrichments than any of the receptor-based approaches (Pérez-Nueno *et al.*, 2008).

4.3.6 Clustering and Classifying Diverse CCR5 Ligands

Several earlier computational docking studies have indicated that different CCR5 ligands bind in fundamentally different ways within the CCR5 extracellular pocket. Furthermore, considering that it is very difficult to superpose all the different families of CCR5 active compounds, there is good evidence to support a hypothesis that the known binders belong to two or more groups and that the members of each group bind to the same general region of the extracellular pocket. In order to explore this hypothesis further, Violeta and I developed a simple method of calculating a "consensus" SH molecular shape representation, $\tilde{r}(\theta, \phi)$, by taking the average of the individual SH molecular surface shapes

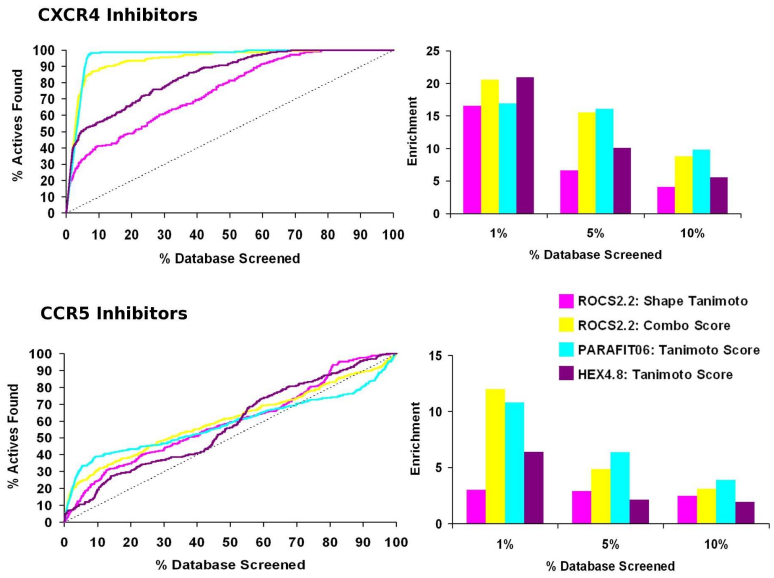

Figure 4.27: VS enrichment curves for the CXCR4 and CCR5 inhibitors. The enrichment curves on the left show the relative ability of the ROCS, ParaFit, and *Hex* programs to identify known inhibitors from a database of inhibitor and decoy molecules for the CXCR4 and CCR5 proteins. The dotted line represents the enrichment that would be obtained if inhibitors were retrieved at random. The bar charts on the right show close-up views of the EFs for the first 1%, 5%, and 10% of the screened databases.

for selected groups of N molecules:

$$\tilde{r}(\theta,\phi) = \frac{1}{N}\sum_{k=1}^{N}\sum_{l=0}^{L}\sum_{m=-l}^{l} a_{lm}^{k} y_{lm}(\theta,\phi). \tag{4.83}$$

However, before computing the average, each molecule in the consensus must first be rotated to minimise the distance between it and the remaining *N-1* molecules. Because these rotations are not known *a priori*, the consensus shape is constructed iteratively as follows. First, all-against-all rotational pair-wise superpositions are calculated in order to find the two most similar surface shapes. Then, the average of these two shapes is taken as the initial seed shape for the consensus, and the remaining *N-2* SH shapes are rotated into superposition with the seed shape. The overall average of all SH coefficients is then computed to give the first estimate of the consensus shape. The consensus average is then refined by superposing the member molecular shapes back onto the average and by recalculating a new average shape. This procedure is repeated until convergence to optimal overlap between each molecule and the consensus shape is reached. This typically requires just three or four cycles. Hence calculating a consensus shape is a quick process.

Figure 4.28(a) shows the consensus shape calculated from the three most active compounds of different scaffold families in the CXCR4 inhibitor database. Figure 4.28(b) shows the consensus

shape computed from all the CXCR4 inhibitors in the database. Visual inspection of these figures shows that the first consensus shape captures rather well the overall shape of the three selected inhibitors, whereas the all-molecule consensus has much less local surface detail, yet still broadly retains the gross features of the member shapes. Figure 4.28(c) shows the consensus shape calculated for the three most active compounds of different scaffold families in the CCR5 inhibitor database. Figure 4.28(d) shows the consensus shape of all the CCR5 active inhibitors. In this case, it can be seen that using all database compounds to construct the consensus query produces a much more spherical average shape than the CXCR4 inhibitors due to the greater number and diversity of compounds in the CCR5 database.

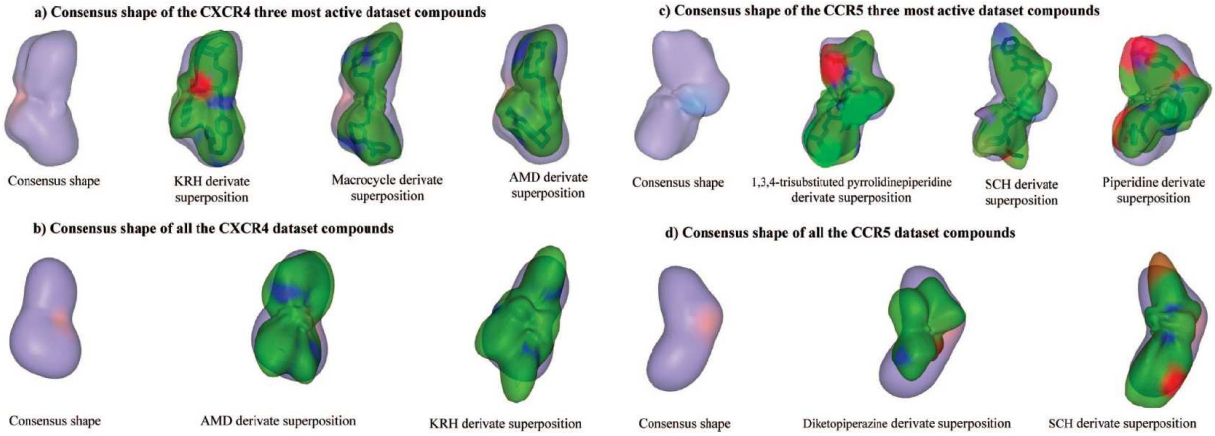

Figure 4.28: CXCR4 and CCR5 antagonist consensus shapes. (a) The image on the left shows the consensus shape calculated from the three most active compounds of different scaffold families in the CXCR4 inhibitor database. The following three images show the superpositions of these compounds onto the consensus. (b) The consensus shape calculated from all CXCR4 database actives, and example superpositions onto the consensus of two randomly selected compounds. (c) On the left, the consensus shape calculated from the three most active compounds of different CCR5 database scaffold families. On the right, the superpositions of these compounds onto the consensus. (d) Consensus shape calculated from all CCR5 actives, along with example superpositions onto the consensus of two randomly selected actives.

Figure 4.29 shows the performance of the CXCR4 consensus VS queries compared to docking-based and shape-based VS using a single high affinity ligand (AMD3100). This Figure shows that the consensus shape queries give higher AUCs than the other approaches, although the single-ligand ParaFit query also performs well. As might be expected from consideration of Figure 4.28, the three-ligand consensus performs considerably better than the all-ligand consensus, due to the high degree of smoothing and loss of surface detail in the all-ligand shape. On the other hand, considering the very good performance of the single high affinity ligand and the marginally superior performance of the three-ligand query suggests that all three ligands share highly similar shapes (as confirmed by Figure 4.28(a)) which probably all bind in similar way within the CXCR4 pocket.

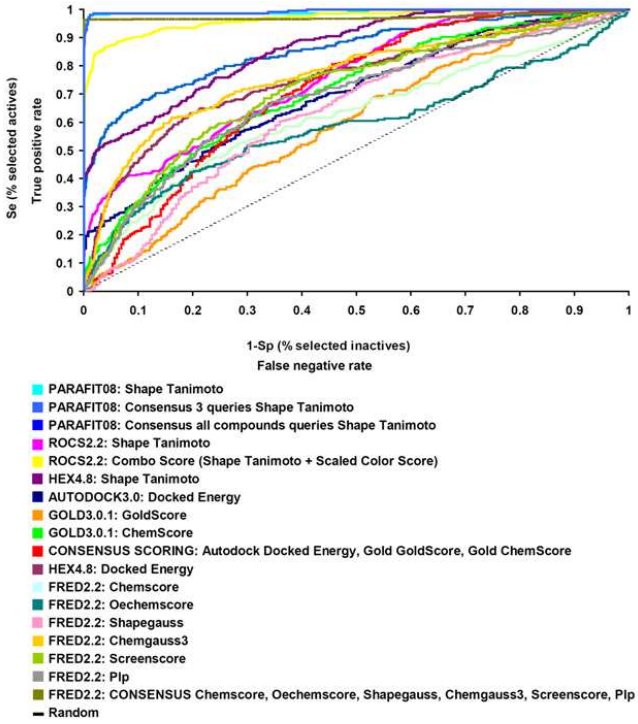

Figure 4.29: ROC plot analyses of ligand-based and receptor-based VS using AMD3100 and consensus pseudo-molecule shape queries against the CXCR4 inhibitor database.

For the more difficult problem of understanding the binding modes of the diverse CCR5 ligands, we supposed that if all of the members of a group bind within the same region of the receptor pocket, then they should all share a significant degree of shape similarity, and that it might be possible to describe these similarities by constructing a consensus “pseudo-molecule”. Indeed, because most of the ligand conformations had been calculated using molecular modelling software, and that some of these conformations might therefore be incorrect, we further supposed that calculating a consensus shape might smooth out some of these errors and possibly provide a better query shape with which to perform ligand-based VS. However, it is not clear *a priori* which actives might belong to which group, nor is it clear how best to superpose all of the molecules within a hypothesized consensus group. Hence we decided to proceed iteratively by performing an initial round of Ward’s hierarchical clustering using conventional chemical descriptors (for full details, see Pérez-Nueno *et al.* 2008) to obtain a total of ten clusters. SH consensus surface shapes were calculated for each cluster, and an all-against-all SH comparison of each consensus surface was calculated using ParaFit. The resulting pair-wise Tanimoto similarity coefficients were then used in a further round of Ward’s hierarchical clustering. Figure 4.30 shows a dendrogram of the resulting super-clusters in which the initial ten consensus shape surfaces are clustered to give four main groups, A, B, C, and D. The members of

these four groups were re-aligned to form four “super-consensus” (SC) pseudo-molecular SH surface shapes. Figure 4.31 shows the 3D molecular overlays, the SH shapes of the ten initial clusters, and the four SC shapes calculated from the clustered consensus surfaces.

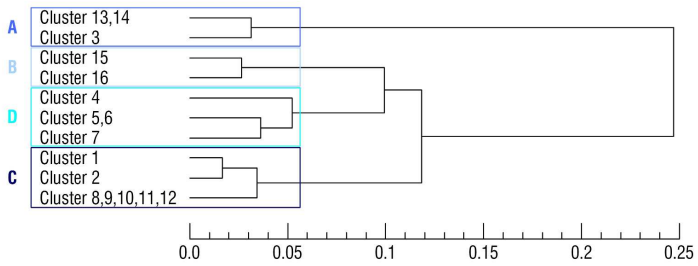

Figure 4.30: Dendrogram of the ten initial CCR5 antagonist groups clustered using Ward’s clustering of SH distances between the consensus surface shapes of each group. Four main super-consensus groups, labelled A, B, C, and D, are identified.

Figure 4.32 shows the VS results obtained using the four SC shapes as database search queries. SC C gives the best overall VS performance with an AUC of 0.91. It is perhaps not surprising that this SC query performs very well because it includes the three most active compounds in the database and also a large number of other actives (i.e. 184/424) with similar shapes to the 4-piperidine derivatives, SCH derivatives and 1,3,4-trisubstituted pyrrolidinepiperidine derivatives. The SC A query (87/424 actives) also performs rather well with an AUC of 0.79, and the SC D query (84/424 actives) performs reasonably well (AUC=0.63). However, the ROC plot for SC B shows that this query exhibits good sensitivity and selectivity in the first percentages of the database screened, but the overall AUC is low (0.41) because the database contains relatively few members of the two SC B families (i.e. a total of only 69/424). However, if the members of clusters B and D are grouped together to form a single SC, as might be suggested by the dendrogram in Figure 4.30, the screening performance becomes essentially random (AUC=0.51). Thus, despite the small populations of these two groups, their members have significantly different overall shapes, and they should be classified as two distinct structural groups for VS purposes. Performing a similar exercise with other combinations of SC clusters shows similar but less dramatic reductions in AUCs compared to the AUCs of the unmerged clusters (details not shown). This behaviour suggest that the CCR5 inhibitor families may be clustered into no fewer than four main groups.

Because *Hex* primarily uses a shape-density representation for its protein-protein docking calculations, it was immediately possible to apply SPF rigid body docking to locate the four SC pseudo-molecular shapes into the CCR5 extracellular pocket. Although protein-protein docking calculations

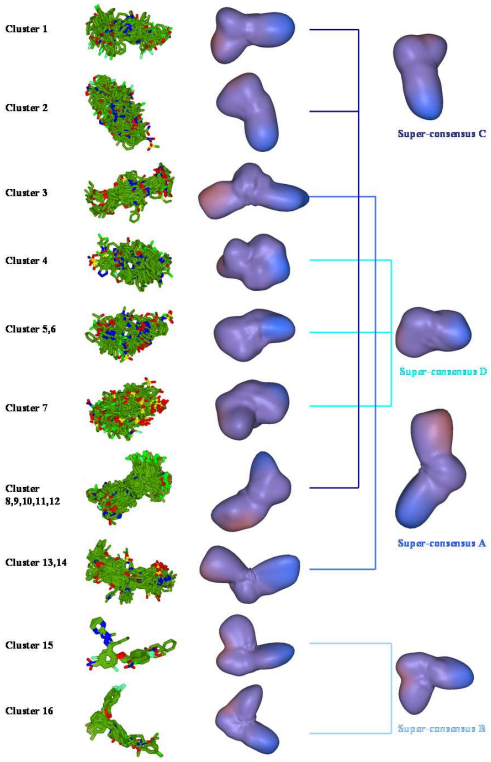

Figure 4.31: Molecular superpositions and consensus shapes of the ten Ward's clusters used to calculate the final four CCR5 super-consensus shapes.

usually generate multiple false-positive orientations, in the present highly constrained case there are very few ways in which the SC "ligands" can fit satisfactorily into the pocket, and in fact only three possible binding sites within the CCR5 trans-membrane (TM) helix regions which could accommodate the four SC shapes were found. These are shown in Figure 4.33. These SC-based docking predictions are consistent with existing experimental data (see Pérez-Nueno *et al.*, 2008, and references 11, 13, 14, and 16 therein for full details).

In order to confirm that the SC queries are properly matched with their predicted target sites, the three proposed binding sites were each treated as if they were separate targets for docking-based VS using rigid body docking of the corresponding SC pseudo-molecules. In other words, when docking to Site 1, compounds belonging to SC A were treated as actives, and compounds belonging to SC B, C, and D were treated as inactives. Similarly, when docking to Site 2, compounds belonging to SC C were treated as actives, and compounds belonging to SC A, B, and D were treated as inactives. Similarly for Site 3, compounds belonging to SC B and D were treated as actives, and compounds belonging to SC A and C were treated as inactives. Figure 4.34 shows the docking VS performance for each of the three proposed CCR5 binding regions. Comparing Figures 4.34 and 4.32, it can be

observed that docking VS onto Sites 1, 2, and 3 (AUC=0.83, 0.96, and 0.85, respectively) improves the SC A, C and B/D shape matching AUCs (AUC=0.79, 0.91, and 0.41/0.63, respectively). Given that SCs A and C already give good shape matching enrichments, reassigning the B and D members as inactives only marginally improves the corresponding AUCs. However, treating the large set of C and A members as inactives for Site 3 gives much higher AUCs for the SC B and D queries, which clearly supports the notion that the CCR5 antagonists bind to at least three main sites within the extracellular pocket.

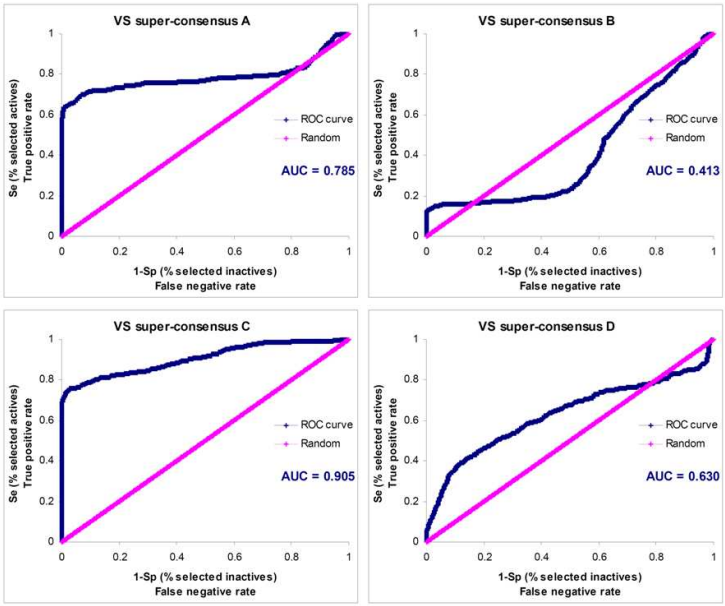

Figure 4.32: ROC plot validation of the CCR5 inhibitor SC pseudo-molecules.

The results of this study showed that SH consensus shapes can provide effective 3D query structures for shape-based VS. For the CXCR4 and CCR5 ligands studied here, our results show that well-chosen consensus shape queries can give better (CXCR4) or significantly better (CCR5) virtual screening enrichments than conventional single-molecule VS queries. However, for CXCR4, these results are nonetheless broadly similar to the basic ParaFit one-molecule shape matching approach because the inhibitors for this target share rather similar molecular shapes which individually match quite well the selected query shape. For CCR5, which has a much larger and more diverse set of inhibitor families, the SC family C and the SC all-family queries both give very good overall VS performance. However, this seems to be at least partly because a high proportion of all scaffold families cluster into the family C super-consensus grouping. Hence, by construction, the spherical harmonic consensus shape derived from these family members provides a single representative pseudo-molecular shape which recognises well many of the individual member structures.

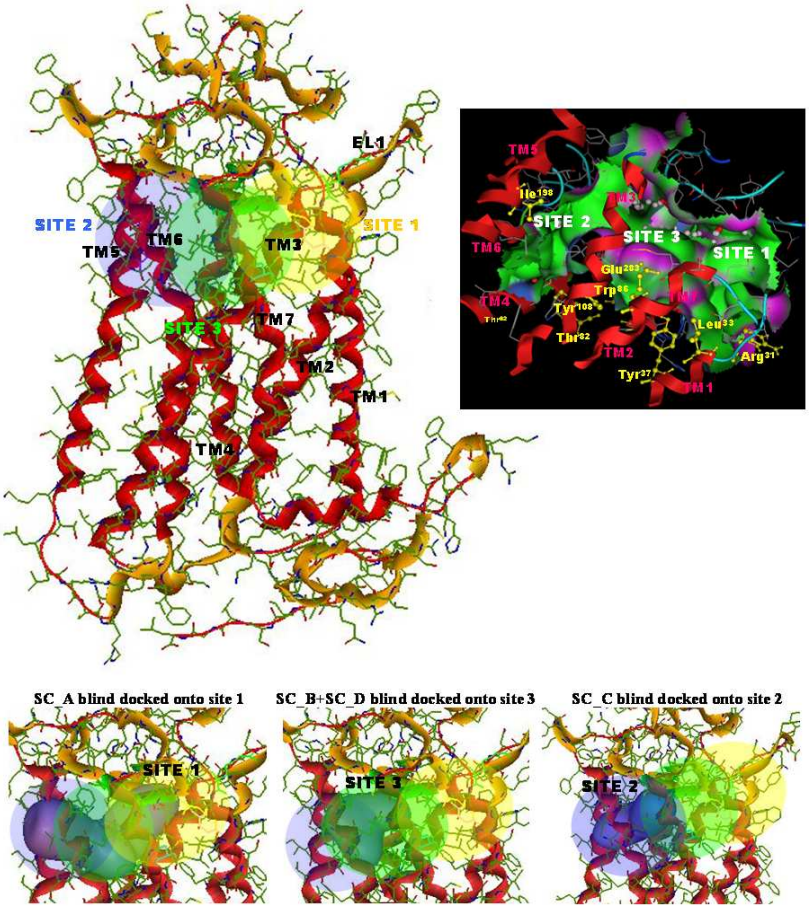

Figure 4.33: CCR5 binding pocket sub-sites proposed by the consensus VS and docking results. Here, the SC A pseudo-molecule is docked onto Site 1, the SC C pseudo-molecule is docked onto Site 2, and the SC B and SC D pseudo-molecules are docked onto Site 3, which overlaps the SC A and SC C sub-sites.

Regarding the more challenging problem of understanding how so many diverse inhibitor families might bind within the CCR5 pocket, our consensus shape-based approach provides a straightforward way to identify clusters of inhibitor families from a large set of known actives which is broadly consistent with previous computational models and current experimental data. Nonetheless, the only completely reliable way to verify the validity of docking-based predictions is through comparison with known crystallographic structures, but unfortunately such gold standard references are not available for CXCR4 and CCR5. Hence any comparison with previous docking studies can, at best, serve only to add further support to the original prediction. On the other hand, for practical purposes, an unbiased and objective way to validate a computational prediction even when no crystal structure is available is to test its utility in the context of VS. The VS results obtained here using four SC shape-based similarity and docking queries give significantly enhanced VS enrichments compared to single-molecule

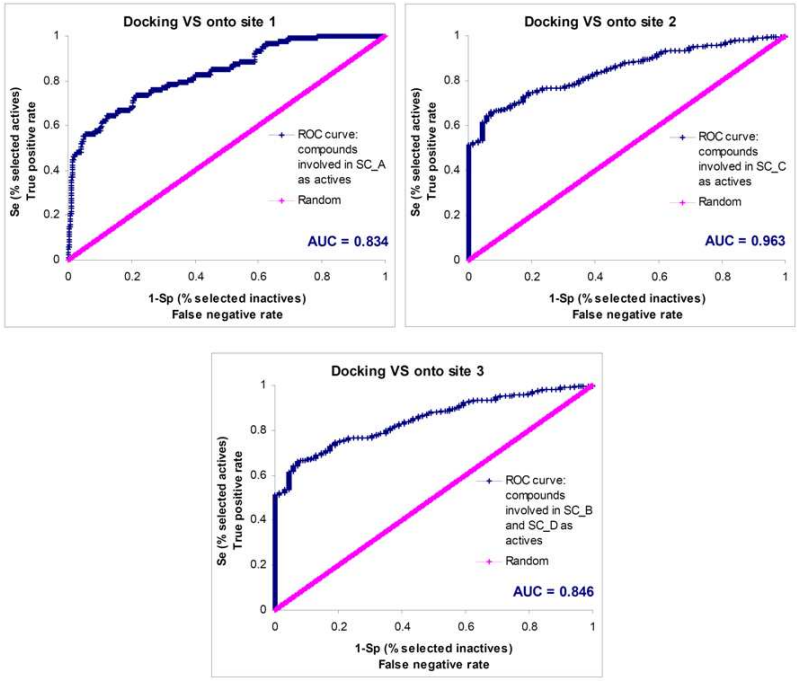

Figure 4.34: ROC plot validation of docking VS onto the three identified CCR5 sub-sites for the CCR5 antagonists.

queries. These results lend strong support both to the validity of the notion of SC structures, and to the hypothesis that the members of these SC clusters bind within at least three main sites in the CCR5 extracellular pocket.

Chapter 5

Summary and Perspectives

5.1 Summary

The preceding chapters have introduced the mathematical machinery with which to calculate SH and SPF representations of the shape and physical properties of proteins and other small molecules. It has been shown that the SPF representation leads to a novel and very efficient way to perform exhaustive protein-protein docking calculations using high order rotational FFT correlations. This approach has been implemented and explored in the *Hex* docking program, which has become the principal vehicle and practical test-bed for much of my research. *Hex* performs respectably well on many of the protein docking benchmark examples, and it has given good results on several of the blind CAPRI docking targets. *Hex* is at least one order of magnitude faster than conventional Cartesian grid-based docking correlation approaches, and it is now one of the most widely used academic protein-protein docking programs in the world. Similarly, it has been shown that the mathematically simpler SH surface representation provides a powerful way to carry out very fast clustering and virtual drug screening of small-molecule databases. The ParaFit program is currently marketed by Cepos Insilico Ltd. as one of the fastest virtual screening tools currently available.

5.2 Future Challenges

Naturally, greater and greater speed is always desirable, and this is especially true for highly interactive graphical applications. On the other hand, in science there is nearly always a need for greater accuracy, and one should always be willing to exchange any gain in speed for more sensitive and more accurate calculations. Furthermore, in the life sciences, there is also an on-going need to be able to apply scientific software to larger and larger biological systems. The recent exponential growth in experimentally determined genomic and protein structure information at multiple levels of resolution and a similar expansion in small-molecule chemical databases means that there are now tremendous

opportunities to exploit protein and chemical structure information for scientific and therapeutic purposes. My on-going objective is to help biologists and medicinal chemists to exploit fully this growing wealth of data by continuing to develop novel computational techniques to represent and manipulate complex biological and chemical molecular structure information.

But the recent explosion of data also means that we now need to raise our sights and aim at much bigger scientific targets and challenges. Instead of being content trying to dock pairs of proteins one by one, we should now be aiming to dock hundreds proteins in order to assemble and simulate large nanometre-scale macromolecular assemblies such as the trans-membrane nuclear pore complex (NPC), and molecular motors such as bacterial flagella and eukaryotic cillia, for example. We should even be aiming to probe the exquisitely sensitive and selective protein-protein recognition process itself by cross-docking thousands of pairs of proteins in order to simulate and verify the large volumes of PPI data produced by high-throughput TAP-MS and Y2H experiments. Similarly, we should be aiming to search vast virtual compound databases for novel drug molecules to screen potential new drug candidates against multiple protein structures in order to predict unexpected or unwanted side-effects, for example. Of course, these are obviously very grand goals. And equally obviously, I am not the first one to have thought of them. But in my opinion, setting high targets is usually a good way to achieve at least an acceptable outcome. The following sections briefly describe some of the more modest directions in which I will continue my research efforts under my current ANR grant, and some possibly more ambitious ideas which I would like to develop.

5.3 Incorporating Knowledge-Based Potentials in Protein Docking

I am continuously working to improve the state of the art in FFT-based docking and to make new developments available in the *Hex* docking software. The novel 5D polar Fourier method of scanning the rigid-body docking space which I recently developed (Section 4.2) is at least one order of magnitude faster than conventional 3D Cartesian FFT docking approaches, and offers several avenues for future development. For example, as the number of known protein complexes continues to grow, so-called knowledge-based protein-protein interaction potentials will become increasingly more accurate and reliable (Ritchie, 2008). However, because such potentials are usually expressed as a sum of contributions from multiple atom or residue types (Kozakov *et al.*, 2006), they are costly to calculate using conventional 3D FFT techniques. Nonetheless, while developing the 5D FFT expression, I showed that the contributions from each term in a knowledge-based potential may be summed *before* performing the rigid body FFT search. In other words, the cost of calculating multi-term potentials in a 5D FFT will be only marginally more than that of calculating simple shape-based correlations. I am collaborating with Dima Kozakov and Sandor Vajda at Boston University to use the 5D FFT approach to calculate the accurate multi-term “Decoys as Reference State” (DARS) protein-protein potential (Kozakov *et al.*, 2006). This novel approach will be able to scan the enormous 6D search space much

more rapidly and reliably than is currently possible. Combining our two state of the art approaches will place our two groups several years ahead of the field.

5.4 Modelling Protein Flexibility During Docking

The project with Diana Mustard to simulate protein flexibility during docking used an essential dynamics approach to generate multiple protein "eigenstructures" to represent snap-shots of the allowed conformational space (Mustard & Ritchie, 2005). We found that docking such conformational snapshots gives promising results but is computationally expensive. More recently, Rueda *et al.* (2009) showed that using protein conformational snapshots from NMA consistently improved the quality of their protein-ligand docking simulations. May and Zacharias (2008) published some very promising protein-protein docking results based on a GNM approach, which models a protein as an elastic network of harmonic springs connecting the backbone C_α atoms. One advantage of GNMs is that all the information necessary for the eigenvector analysis may be derived directly from the Hessian interaction matrix, thus avoiding the need to generate multiple pseudo-random conformations. Nonetheless, there is still scope for improvement. For example, one drawback of the current GNM approach (and of eigenvector approaches in general) is the need to diagonalise large matrices, the cost of which scales as $O(N^3)$. Hence existing approaches have so far been limited to using only the coordinates of the C_α atoms, and have had to perform a single diagonalisation for each protein before the docking run.

In the Eigen-Hex project, we will apply a fresh GNM analysis for each putative rigid body docking pose. Thus each eigenvector analysis will take into account the accessible fluctuations of each protein in the context of the constraints presented by its docking partner. Although this approach will involve many matrix diagonalisations, we will reduce the computational expense by using the "building block" approach of Tama *et al.* (2000) in which a protein is subdivided into small blocks of up to six amino acids per block, each of which is treated as a rigid rotational-translational block (RTB). Tama *et al.* showed that the normal modes obtained from this approximation are almost as accurate as using one amino acid per block, yet the computational cost is dramatically reduced (Tama & Sanejouand, 2001). By using the resulting eigenvectors to generate new putative conformational poses and rapidly energy-minimising each new pose using soft molecular mechanics (Ritchie, 2003), each rigid body pose will be flexibly steered into a locally optimal binding mode. However, even using efficient RTB eigenvectors (and perhaps also sparse matrix diagonalisation techniques), this approach will be computationally intensive because each promising conformation will need to be re-scored using a full molecular mechanics force field. Hence the calculations will be distributed over multiple processors. I am currently investigating these ideas with Vishwesh Venkatraman who recently joined my group as a postdoc funded by my ANR grant.

5.5 Automating Data-Driven Protein-Protein Docking

As structural genomics initiatives continue to populate the space of protein 3D structures, using structural database systems to perform docking by homology (which one might describe as “docking using case-based reasoning”) will obviously become an increasingly powerful approach to predicting protein interactions. Although the PDB currently contains only a very small fraction of all possible protein-protein complexes, several groups have recently developed PPI databases (for reviews, see e.g. Mathivanan *et al.* 2006 and Ritchie 2008), and these are becoming important assets with which to predict the structures of protein complexes. The growing number and coverage of such databases now offers the prospect of developing automated ways of transferring knowledge of existing protein-protein interactions to unknown but homologous cases. Even in cases where there are no direct structural homologues of a docking target system, it is still extremely useful to be able to incorporate knowledge about key interaction residues into the docking protocol. Indeed, knowledge of even a single interaction residue can help focus the scope of a docking calculation, and dramatically improve the quality of the result (Ritchie, 2008).

If one considers the various approaches used by the different CAPRI participants, it is clear that several predictors have become skilled at finding and understanding prior knowledge about the targets from the literature. However, this is a time-consuming and error-prone activity, and it would be desirable to develop automated data-mining approaches which can emulate the experts’ knowledge acquisition steps for data-driven docking. I would like to develop a generic way to incorporate knowledge from external data sources into docking calculations by incorporating ambiguous interaction restraint (AIR) terms (Nilges, 1995) into the docking energy function, for example. However, it is not yet clear how best to translate activities carried out by human experts into simple rules that can be executed and applied within a database.

The ANR funded “KDD-Dock” doctoral thesis project of Anisah Ghoorah which I am supervising in collaboration with my LORIA colleagues Marie-Dominique Devignes and Malika Smaïl-Tabbone represents a first step towards applying formal Knowledge Discovery in Database (KDD) techniques to the task of extracting information from existing PPI databases to help guide protein docking calculations. KDD often involves processing huge volumes of data using a variety of data mining techniques in order to extract useful and reusable rules or “knowledge units.” Common data mining techniques include lattice-based classification, frequent item-set search, and association rule extraction approaches (Napoli, 2005). However, it is often first necessary to collect the data into a single large table in order to apply data mining. This can be a difficult task if the data is distributed over several tables or data sources because it requires squeezing the data into a single regular table (e.g. by making a database view from one or more joins or aggregations, for example). On the other hand, relational data mining (RDM) approaches can perform data mining on multiple tables, but they cannot easily be applied to large datasets due to their algorithmic complexity. Nonetheless, it is possible to combine

RDM with more efficient conventional approaches (Helma *et al.*, 2004; Muggleton, 2005; Phuong & Ho, 2005). Hence the overall aim of this project is to apply KDD techniques to existing PPI databases to discover simple rules about protein interfaces which could be transformed into AIRs for docking calculations.

In the longer term, I also believe that improving our ability to identify PPI information from the literature and using hidden markov models (HMMs) to detect protein interface sites will be further useful strategies for constraining and guiding docking calculations. I would therefore like to develop collaborations along these lines with my Orpailleur colleagues.

5.6 Performing 3D Protein Structure Alignment and Classification

Although Lazaros Mavridis began work on the ANR funded "3D-Blast" project only in March 2009, the early results reported in Sections 4.1.2 and 4.1.3 are extremely promising. We have already shown that the SPF approach is sufficiently sensitive to superpose and cluster protein structures in very good agreement with CATH classification, which is one of most widely accepted protein structures reference resources. We have also shown that it should soon be possible to perform fast sequence-independent structural queries on the entire CATH database in interactive time by using rotationally invariant and rotational correlation searches in tandem. Although it will be very interesting to perform large-scale clustering experiments on the entire CATH database, we do not propose that our SPF-based clustering approach should completely replace CATH. Rather, we see it as providing an independent and objective way to complement the time-consuming expert classification currently used in CATH. We also expect it could help resolve difficult or ambiguous cases, and provide a way to identify unexpected or novel similarities that might exist beyond the "twilight zone" of conventional sequence alignment. Extending our approach to provide substructure querying and matching could become useful additional capabilities with which to perform more detailed structure-function analyses. For example, as part of his Masters project at the LORIA, Emmanuel Bresso is currently investigating how this approach might be used to perform a large-scale study of the structural determinants of phosphorylation sites on protein surfaces.

5.7 Exploring Visuo-Haptic Steered Protein Docking

One of the main challenges in protein docking today is to distinguish near-native binding modes from a list of highly plausible false-positives (Ritchie, 2008). Although progress continues to be made in developing automated docking approaches, there has been remarkably little research on how human skills and abilities might also be exploited to help improve docking results. Experience from the CAPRI blind docking experiment suggests that human skill can still make a very important contribution to the quality of a set of docking predictions. Many CAPRI participants, and no doubt most laboratory re-

searchers, devote a considerable amount of time viewing and analyzing candidate docking solutions using 3D molecular graphics tools. Typically, the feasibility of each docking prediction is assessed visually to consider the snugness of fit between the calculated interaction surfaces. However, beyond developing better energy-based scoring functions, it is not clear how this activity could be automated. Nonetheless, Gillet *et al.* (2005) demonstrated that the ability to feel and manipulate tangible plastic molecular models provides an unparalleled level of intuition and information about the shapes and properties of macromolecules. Unfortunately, creating 3D physical models of molecules ("3D printing") remains a slow and expensive process. On the other hand, some recent virtual reality work on modelling protein-ligand interactions has shown that the interaction forces between proteins and small ligands can usefully be simulated and experienced physically using haptic feedback techniques (Nagata *et al.*, 2002; Wollacott & Mertz, 2005). Therefore, it would be very worthwhile to explore such an approach in the context of protein-protein docking.

Uniquely, the *Hex* docking software already has a built-in stereographic molecular visualisation capability with which to view the calculated docking poses, but like most conventional molecular graphics software it lacks haptic feedback. Nonetheless, the interactive nature of the existing software means it will be relatively straightforward to add this capability and to explore its utility. The basic idea is that the user would view and manipulate before his eyes the proteins almost literally as if holding one protein in each hand. Twisting the wrists (pitch and roll motions) would be transformed into Euler rotation angles with which to rotate the proteins. Moving the hands together or apart would control the Cartesian positions of the molecules. This would provide a much more natural way of manipulating and inspecting docking poses than the conventional 2D mouse control that is currently implemented in *Hex*, for example. Additional visual feedback would be provided by colour-coding the protein surfaces according to docking energy and steric clashes, for example. If key interaction or restraint residues have been identified, these will be highlighted graphically in data-driven docking runs. Exploring interactive protein docking in this way would fit very well with the above plans regarding data-driven docking and developing fast energy and scoring functions. For example, gradient-based techniques can be used to calculate the forces necessary for inertial haptic feedback, and it would be very interesting to explore ways of visualizing interactively energy funnels around a putative docking pose. It is worth noting that the LORIA has excellent virtual reality facilities and several teams are working in related areas. Hence further synergies and innovations could emerge through a project of this kind.

5.8 Developing SH Consensus Shape Virtual Screening

Understanding how proteins interact is crucially important for understanding the molecular mechanisms of disease. For example, therapeutic drugs often work by modulating or blocking PPIs, and therefore PPIs represent an important class of drug target (González-Ruiz & Gohlke, 2006). Therefore, searching chemical databases to find small ligand molecules which might bind to specific pro-

teins and modelling those interactions computationally will become increasingly important strategies for structure-based drug discovery (Richards, 2002; Congreve *et al.*, 2005). Although it is clearly desirable to know the three-dimensional structure of the protein target, this is often not possible especially for large trans-membrane protein structures which are very difficult to crystallise, for example.

Nonetheless, in ligand-based VS, the structure of the protein target is generally not known, and the overall approach is to use knowledge of existing high affinity ligands in order to find or design further similar molecules which might bind to the protein target in a similar way. Indeed several studies to compare receptor-based docking and ligand-based shape matching VS approaches, have shown that ligand-based methods perform at least as well as and often better than docking (Hawkins *et al.*, 2007). However, despite the relative success of ligand-based VS, there remain the confounding problems of how to choose the initial query compounds and which of their conformations should be used (Hawkins *et al.*, 2007). Additionally, it is increasingly the case that pharmaceutical companies have developed multiple ligands for a given target. However, traditional shape matching approaches normally use just one conformation of a compound as the query, but it is not known *a priori* if this is the correct query to use to screen an entire database. For example, other compound families could also be active for the same target but they might only be found in the database if a different query conformation is used. In other words, conventional VS assumes there is only one binding mode for a given protein target. This may be true for some targets, but it is certainly not true in all cases. Several recent studies have shown that some protein targets bind different ligands in different ways, e.g. CCR5 (Kellenberger *et al.*, 2007), CXCR4 (Wong *et al.*, 2008), CDK2 (Wong *et al.*, 2006), HIVRT (Lewis *et al.*, 2003), FXA (Taha *et al.*, 2005), and LXR (Williams *et al.*, 2003). Hence there is a need to develop new VS approaches which can detect such cases and which can associate specific sub-sets of ligands with their corresponding receptor binding sites.

As summarised in Section 4.3.5, the work with Violeta Pérez-Nueno on SH-based VS demonstrated that the SH representation provides an effective and rapid way to identify ligands which are globally similar to a given query molecule, and that the SH approach implemented in ParaSurf and ParaFit gives comparable results to the industry-standard ROCS Gaussian superposition program (Grant *et al.*, 1996). We also showed that by using the SH representation, it is straightforward to construct the average or “consensus” shape of a group of molecules by calculating the average of their coefficient vectors, and that clustering such consensus-shapes indicates that the CCR5 ligands may be classified into four main families which appear to bind within three main sub-sites in the CCR5 extra-cellular pocket. Because we achieved very good results for the CCR5 and CXCR4 systems, we wish to develop and extend the consensus clustering approach and apply it to other protein targets. Consequently, in August 2009 Violeta and I submitted an application for a Marie Curie Intra-European Fellowship which would allow her to develop these ideas at the LORIA.

5.9 Implementing FG-Based Protein-Ligand Virtual Screening

As well as being used to represent the shapes of small molecules, SH representations have also been used successfully to characterise the surface shapes of protein binding pockets (Cai *et al.*, 2002; Morris *et al.*, 2005), and to perform high-throughput receptor-based VS (Yamagishi *et al.*, 2006). However, it is not yet clear whether SH surface representations can accurately match ligand shapes to protein pockets because, mathematically, the surface envelope representation does not help solve the translational part of the matching problem. In my opinion, one must employ more sophisticated 3D SPF representations, similar to those used in *Hex*, in order to provide a higher level of sensitivity in a HTVS filter whilst avoiding the need for expensive HPC facilities. Furthermore, because it has been suggested that 3D ligand shape matching may actually be superior to docking for VS purposes (Hawkins *et al.*, 2007), there are considerable opportunities and potentially high rewards in developing SPF-based approaches for ligand-based HTVS filters.

Although the GL radial basis functions described here work very well for protein-protein docking, I believe the most computationally useful way to represent and compare the 3D shapes of small molecules is to use SPF expansions in which the existing GL radial basis functions are replaced by the Gegenbauer polynomials. There are two main reasons why I believe the Gegenbauer polynomials will provide a more appropriate basis set with which to develop new 3D receptor-based *and* ligand-based HTVS techniques. Firstly, these polynomials do not have an exponential decay factor like the GL polynomials, but instead their natural domain is the unit hypersphere. This means that with a suitable choice of scale factor, the Gegenbauer polynomials will allow a given level of detail to be encoded more compactly using lower order expansions than would be the case with the GL functions. Secondly, because an addition theorem exists for the Gegenbauer polynomials (Srinivasan *et al.*, 2005), it should be possible to calculate the effects of translations as well as rotations by transforming only the original expansion coefficients, as is currently the case with the existing GL basis functions. As far as I know, there does not exist a translation theorem for the Zernike polynomials. Therefore, despite promising recent results with rotationally invariant Zernike shape representations (Mak *et al.*, 2008; Sael *et al.*, 2008; Venkatraman *et al.*, 2009), the Zernike basis would seem to be less attractive than the GL or Gegenbauer bases. It is worth noting that so-called rotationally invariant descriptors may be obtained trivially from all such orthonormal basis set expansions simply by summing the magnitudes of the expansion coefficients (Mavridis & Ritchie, 2009). In any case, the above considerations are important because the computational cost of calculating the initial SPF expansion coefficients scales in the order of $O(N^3)$, and the cost of rotating and translating such representations broadly scale as $O(N^4)$, and $O(N^5)$, respectively, where N is the polynomial order of the expansion. Therefore, by choosing the type of radial expansion and scale factor wisely, one can reduce the expansion order and hence significantly speed up shape matching and docking calculations without sacrificing resolution.

It is envisaged that the new 3D-Snap software will be able to operate in either a ligand-based or

receptor-based screening mode, or both. In other words, if the input is a small molecule, the program will search its database for similar molecules using 3D rotational superposition correlations. If the input is a protein target, the program will search for molecules which can dock into a specified binding site on the target protein. In this case, the binding site will be indicated by providing the coordinates and radius of a dummy atom on the protein, for example. If the input is an existing protein-ligand complex, the program will search its database for similar ligands and it will then dock them into the binding site implied by the location of the ligand. It will be possible to search the database using different filtering modes. For example, for any given query shape, ultra-fast rotationally invariant comparisons will be used to eliminate almost instantly a large proportion of the database. More sensitive matching will then be performed using explicit FFT rotational correlations. When appropriate, receptor-ligand docking correlations will only be applied to candidate ligands that pass the initial shape-based filters. Hence it will only be necessary to perform full docking calculations on a relatively small subset of the database.

With a given ligand binding site, the current version of *Hex* can perform rigid-body docking in around one CPU-minute because the calculation largely reduces to a rotational correlation in which the ligand spins within the binding site, and only a limited translational search is required to complete the docking manoeuvre. However, *Hex* is not designed for accessing a database or for working with a filtering pipeline, and it would be inappropriate to overload the structure of that software. Hence the new program, 3D-Snap, is proposed. Nonetheless, much of 3D-Snap will be implemented by adapting and re-using a lot of low level software from *Hex*. By using new FG-based shape representations of the ligand and only a relatively small region of the protein centred on the binding site, it will be possible to perform very fast but accurate matching/docking correlations. The overall approach is shown schematically in Figure 5.1. From previous experience it is expected that each docking correlation will require just a few CPU-seconds.

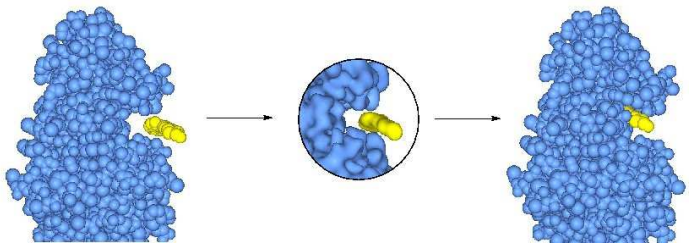

Figure 5.1: Schematic illustration of the proposed protein-ligand FG docking correlation calculation. Left: a fibroblast growth factor receptor kinase (FGFRK) domain with an ATP analogue ligand near the ligand binding site (PDB code: 1AGW). Centre: FG representations of the ligand and the local region around the FGFRK ligand binding site. Right: the calculated FGFRK-ligand complex. Because the ligand binding site is normally known *a priori*, it is not necessary to scan the entire protein surface during docking. Instead, a fast SPF correlation may be performed by spinning and translating the ligand within a small spherical region local to the binding site (centre). This illustration is drawn using GL functions instead of the proposed hyperspherical FG basis functions.

Although I have quite a lot of experience of calculus and special function theory, it will likely involve a non-trivial amount of work to develop the necessary translation expressions to fully exploit the proposed FG approach, and it is difficult to predict how much time this might require. I would not normally expect a postdoc to be able to undertake this part of the project. Nonetheless, it should be noted that neither the 3D-Blast nor 3D-Snap projects critically depend on the FG basis functions for their success. Thus, software development and testing of both of these projects may proceed using the existing GL basis functions from *Hex*.

This approach will be developed and evaluated using standard HTVS protein targets and ligand datasets, such as those provided by the public DUD (Directory of Useful Decoys) dataset and ZINC database (Irwin & Shoichet, 2005; Huang & Shoichet, 2006). Although the proposed protein-ligand docking calculations would not be as accurate as AutoDock, this approach would provide an additional fast filter that could precede the conventional force-field based protein-ligand docking stage. I believe that by building a filtering and docking pipeline using 3D FG representations and fast 3D and 5D SPF correlation techniques, it will be possible to scan millions of ligands with comparable accuracy to AutoDock in around one day on readily affordable modern GPU hardware (described in more detail below). Successfully achieving this level of performance would revolutionise structure-based drug discovery efforts.

5.10 Harnessing State of the Art Graphics Processors

Recent advances in graphics processor unit (GPU) technology have made available incredible computational power on the desktop at an affordable price. Developments in GPU technology were initially driven by the demands of the computer graphics gaming industries, but many scientific calculations have since been adapted to run on GPUs (Owens *et al.*, 2007). Indeed, it is now possible to perform calculations on a desktop computer that most scientists could only dream of just a few years ago. For example, the current state of the art GPU from nVidia contains 240 arithmetic processor units which together can deliver almost 1000 billion floating point operations per second (i.e. 1 Teraflop). This corresponds to over 100 times the computational power of a conventional desktop workstation. Until recently, it required considerable skill and specialist hardware knowledge to write programs for GPUs. However, with the advent of the SIMT (simultaneous instructions on multiple threads) hardware model, and with new software development tools such as Brook (Buck *et al.*, 2004) and CUDA[1], it is now much easier to deploy scientific software on GPUs. For example, for MD simulations, Buck *et al.* (2004) achieved nearly a 10-fold speed-up for the Gromacs software using Brook, and Stone *et al.* (2007) reported speed-ups of up to a factor of 36 for NAMD using CUDA. Although there remain significant difference between the Brook and CUDA programming models, it would appear that the

[1] http://www.nvidia.com/.

forthcoming OpenCL[2] specification will soon become the industry standard programming model for cross-platform parallel processing. Naturally we will track any new developments such as OpenCL.

I have recently implemented the 1D and 3D *Hex* docking correlations in CUDA to obtain a speed-up of at least 45 times compared to the same calculation on a single conventional CPU (Ritchie & Venkatraman, 2010; Macindoe *et al.*, 2010).

Furthermore, as part of the HPASSB project, we have extended *Hex* to use up to eight CPUs and two GPUs simultaneously on a single desktop machine. This now allows exhaustive shape-based docking calculations to be performed in a matter of seconds, thus opening the possibility of allowing biologists to perform truly interactive "steered" docking on their desktop. It will allow much more sophisticated models of protein flexibility to be incorporated into the calculations by trading greater speed for higher accuracy. However, despite using efficient computational techniques, the Eigen-Hex and 3D-Snap projects will be computationally demanding. Performing flexible docking in Eigen-Hex will involve cross docking and energy minimising multiple protein conformations. Similarly, performing virtual drug screening on large compound databases will involve comparing and filtering in the order of millions of compounds. Therefore, developing efficient algorithms using modern parallel programming models to exploit state of the art hardware will be an import aspect of future developments in structural systems biology.

5.11 Modelling Macromolecular Assemblies

Developing high performance algorithms will also be very important in large-scale macromolecular modelling projects. Figure 5.2 shows the crystal structures of the seven component Arp2/3 complex (Robinson *et al.*, 2001) which is responsible for initiating actin polymerisation in eukaryotic cells, and the ten component yeast RNA polymerase II elongation complex (Gnatt *et al.*, 2001) which is involved in transcribing DNA into messenger RNA. Intriguingly, Inbar *et al.* (2005) demonstrated that it is possible to build these multi-component structures from the *unbound* component proteins by using a combinatorial assembly technique which requires only pair-wise docking results. It seems that the combinatorial assembly approach works so well because each putative pair-wise arrangement provides substantial steric constraints on how subsequent structures might be added to the emerging complex. In other words, there are sufficient mutual steric constraints to permit a single near-native solution to be identified. This ground-breaking work demonstrated the feasibility of assembling very large multi-component structures. However, further research is required to develop more practical search techniques which do not require an exhaustive combinatorial search.

Although X-ray crystallography is undoubtedly the gold standard structure determination technique, the resolution available in cryo-electron microscopy (cryo-EM) has improved considerably over the years, and is beginning to approach that of crystallography (Stowell *et al.*, 1998). One advantage

[2] http://www.khronos.org/opencl/.

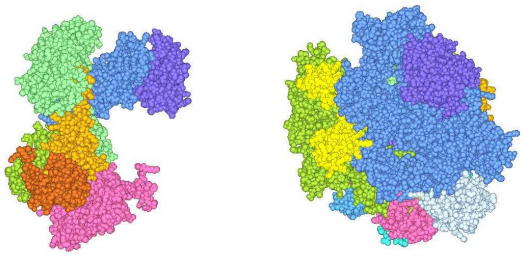

Figure 5.2: Two examples of crystallographically solved multimolecular assemblies. The structure on the left is the seven component Arp2/3 complex (PDB code 1K8K) which is responsible for initiating actin polymerisation in eukaryotic cells. The structure on the right is the ten component yeast RNA polymerase II complex (PDB code 1I6H) which is responsible for transcribing DNA into messenger RNA.

of cryo-EM is that it can provide low resolution structures of very large macro-molecular assemblies which may be difficult or impossible to solve using conventional crystallographic techniques (Frank, 2002a). In other words, cryo-EM can provide the overall shapes of very large assemblies but it is still necessary to fill in the atomic detail in such structures. For example, the structure of the large trans-membrane vacuolar ATPase motor (see Figure 5.3), was recently solved in this way (Muench *et al.*, 2009). FFT correlation techniques are increasingly being used to fit high resolution X-ray protein structures into low resolution cryo-EM density maps (Wriggers *et al.*, 1999; Roseman, 2000; Kovacs *et al.*, 2003). Such fitting or "interior docking" techniques are likely to remain very powerful approaches for determining atomic resolution structures of very large complexes which are unlikely to be solved using crystallographic techniques (Rossmann, 2000; Rossmann *et al.*, 2007).

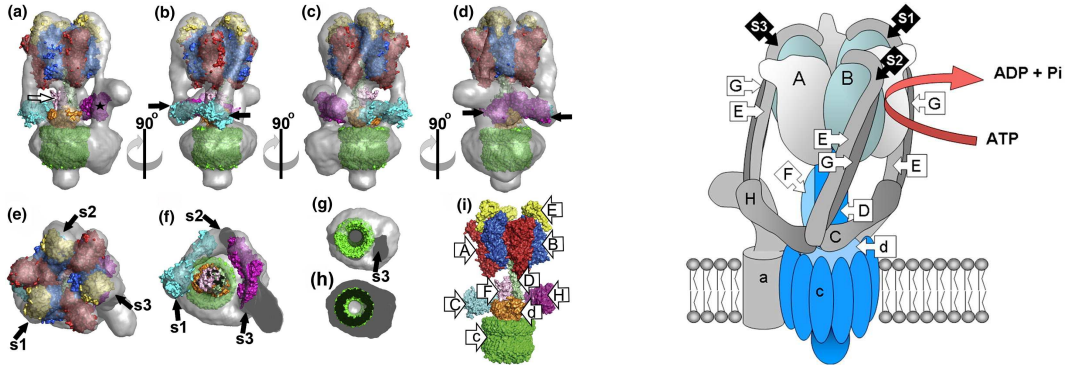

Figure 5.3: Left: illustration of the ATPase molecular motor structure showing the component protein domains docked into the low resolution cryo-EM density map (shown in grey). Right: schematic diagram of the modelled arrangement of the stators (grey), and rotors (blue). These images are taken from Muench *et al.* (2009).

On an even larger scale, Alber *et al.* recently showed that the overall architecture of the very large nuclear pore complex (NPC), comprising a total of 456 proteins, may be modelled by combining

diverse multi-resolution data on the component protein structures and their interactions (Alber *et al.*, 2007a; Alber *et al.*, 2007b). Figure 5.4 shows the predicted positions of the main sub-structures in this remarkable model. This model nicely exemplifies the future need to be able to use hybrid approaches which can integrate diverse sources of experimental data and PPI knowledge to bridge the resolution gap between different experimental structural data collection techniques (Elad *et al.*, 2009; Lindert *et al.*, 2009).

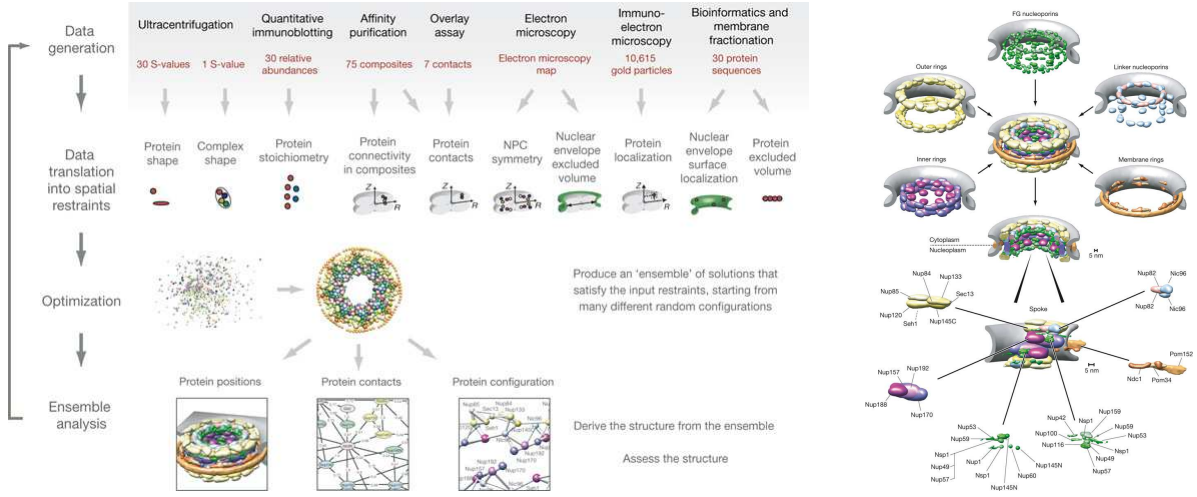

Figure 5.4: Left: schematic overview of the NPC model-building process of Alber *et al.* requiring the integration of diverse experimental data and spatial constraints. Right: localisation of the main sub-structures in the final model. These images are taken from Alber *et al.* (2009a and 2009b).

The ATPase motor and the NPC are just two examples of molecular machines. Improving our ability to model the structural biology of such molecular machines would make an enormous contribution towards understanding dynamical processes within the cell (Stein *et al.*, 2007). As well as helping to understand the fundamental machinery of life, being able to model the structural biology of viruses such as HIV (Ganser-Pornillos *et al.*, 2008), and how they assemble themselves could lead to the development of novel assembly-inhibiting therapeutic molecules (Sticht *et al.*, 2005). I believe the SPF algorithms and techniques presented here will have a useful role to play in the many fascinating future challenges in structural systems biology.

Bibliography

Alber, F., Dokudovskaya, S., Veenhoff, L. M., Zhang, W., Kipper, J., Devos, D., Suprapto, A., Karni-Schmidt, O., Williams, R., Chait, B. T., Rout, M. P., & Sali, A. (2007a). Determining the architectures of macromolecular assemblies. *Nature,* **450**, 683–694.

Alber, F., Dokudovskaya, S., Veenhoff, L. M., Zhang, W., Kipper, J., Devos, D., Suprapto, A., Karni-Schmidt, O., Williams, R., Chait, B. T., Sali, A., & Rout, M. P. (2007b). The molecular architecture of the nuclear pore complex. *Nature,* **450**, 695–701.

Aloy, P., Pichaud, M., & Russell, R. B. (2004). Protein complexes: structure predicition challenges for the 21st century. *Curr. Op. Struct. Biol.* **15**, 15–22.

Aloy, P. & Russell, R. B. (2004). Ten thousand interactions for the molecular biologist. *Nat. Biotech.* **22**, 1317–1321.

Aloy, P. & Russell, R. B. (2006). Structural systems biology: modelling protein interactions. *Nature Rev. Mol. Cell. Biol.* **7**, 188–197.

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., & Lipman, D. J. (1990). Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410.

Amadei, A., Linssen, A. B. M., & Berendsen, H. J. C. (1993). Essential dynamics of proteins. *Proteins: Struct. Func. Genet.* **17**, 412–425.

Arkin, M. R. & Wells, J. A. (2004). Small-molecule inhibitors of protein-protein interactions: progressing towards the dream. *Nat. Rev. Drug Discov.* **3**, 301–317.

Beautrait, A., Leroux, V., Chavent, M., Ghemtio, L., Devignes, M. D., Smaïl-Tabbone, M., Cai, W., Shao, X., Moreau, G., Bladon, P., Yao, J., & Maigret, B. (2008). Multiple-step virtual screening using VSM-G: overview and validation of fast geometrical matching enrichment. *J. Mol. Model.* **14**, 135–148.

Biedenharn, L. C. & Louck, J. C. (1981). *Angular Momentum in Quantum Physics*. Reading, MA: Addison-Wesley.

Bork, P., Jensen, L. J., von Mering, C., Ramani, A. K., Lee, I., & Marcotte, E. M. (2004). Protein interaction networks from yeast to human. *Curr. Op. Struct. Biol.* **14**, 292–299.

Bourne, P. E. & Weissig, H. (2003). *Structural Bioinformatics*. New York: Wiley.

Boys, S. F. (1950). Electronic wave functions I. A general method of calculation for the stationary states of any molecular system. *Proc. Roy. Soc.* **A200**, 542–554.

Bransden, B. H. & Joachain, C. J. (1997). *Introduction to Quantum Mechanics*. Harlow, UK: Addison Wesley Longman.

Brown, F. K. (1998). Chemoinformatics: What is it and how does it impact drug discovery? *Annual Reports in Med. Chem.* **33**, 375.

Buck, I., Foley, T., Horn, D., Sugerman, J., Fatahalian, K., & Hanrahan, M. H. P. (2004). Brook for GPUs: stream computing for graphics hardware. *ACM Trans. Graph.* **23**, 777–786.

Cai, W., Shao, X., & Maigret, M. (2002). Protein-ligand recognition using spherical harmonic molecular surfaces: towards a fast and efficient filter for large virtual throughput screening. *J. Mol. Graph.* **20**, 313–328.

Carbo, R., Leyda, L., & Arnau, M. (1980). An electron density measure of the similarity between two compounds. *Int. J. Quant. Chem.* **17**, 1185–1189.

Carrieri, A., Pérez-Nueno, V. I., Fano, A., Pistone, C., Ritchie, D. W., & Teixidó, J. (2009). Biological profiling of anti-HIV agents and insights into CCR5 antagonist binding using in silico techniques. *ChemMedChem,* **4**, 1153–1163.

Chen, R. & Weng, Z. (2003). A novel shape complementarity scoring function for protein-protein docking. *Proteins: Struct. Func. Genet.* **51**, 397–408.

Cherfils, J. & Janin, J. (1993). Protein docking algorithms: simulating molecular recognition. *Curr. Op. Struct. Biol.* **3**, 265–269.

Chiu, L. Y. C. & Moharerrzadeh, M. (1999). Fourier transform of spherical Laguerre Gaussian functions and its application in molecular integrals. *Int. J. Quant. Chem.* **73**, 265–273.

Condon, E. U. & Odabasi, H. (1980). *Atomic Spectra*. Cambridge: Cambridge University Press.

Congreve, M., Murray, C. W., & Blundell, T. L. (2005). Structural biology and drug discovery. *Drug Discovery Today,* **10**, 895–907.

Cuff, A. L., Sillitoe, I., Lewis, T., Redfern, O., amd J. Thornton, R. G., & Orengo, C. A. (2008). The CATH classification revisited – architectures reviewed and new ways to characterize structural divergence in superfamilies. *Nucleic Acids Res.* **37**, D310–D314.

Danos, M. & Maximon, L. C. (1965). Multipole matrix elements of the translation operator. *J. Math. Phys.* **6** (1), 766–778.

Davies, E. K., Glick, M., Harrison, K. N., & Richards, W. G. (2002). Pattern recognition and massively distributed computing. *J. Comp. Chem.* **23**, 1544–1550.

de Groot, B. L., van Aalten, D. M. F., Scheek, R. M., Amadei, A., Vriend, G., & Berendsen, H. J. C. (1997). Prediction of protein conformational freedom from distance constraints. *Proteins: Struct. Func. Genet.* **29**, 240–251.

Dean, P. M. (1995). *Molecular Similarity in Drug Design*. London: Blackie Academic & Professional.

Dean, P. M. & Callow, P. (1987). Molecular recognition: identification of local minima for matching in rotational 3-space by cluster analysis. *J. Mol. Graph.* **5** (3), 159–164.

Debnath, L. & Bhatta, D. (2007). *Integral Transforms and their Applications*. London: Chapman Hall.

Debnath, L. & Mikusinski, P. (1999). *Introduction to Hilbert Spaces with Applications*. London: Academic Press.

Edmonds, A. R. (1957). *Angular Momentum in Quantum Physics*. New Jersey: Princeton University Press.

Edvardson, H. & Smedby, O. (2003). Compact and efficient 3D shape description through radial function approximation. *Computer Methods and Programs in Biomedicine,* **72**, 89–97.

Egan, J. P. (1975). *Signal Detection Theory and ROC Analysis*. New York: Academic Press.

Eisenstein, M. & Katchalski-Katzir, E. (2004). On proteins, grids, correlations, and docking. *Comptes Rendus Biologies,* **327**, 409–420.

Elad, N., Maimon, T., Frenkiel-Krispin, D., Lim, R. Y. H., & Medalia, O. (2009). Structural analysis of the nuclear pore complex by integrated approaches. *Curr. Op. Struct. Biol.* **19**, 226–232.

Erdélyi, A., Magnus, W., Oberhettinger, F., & Tricomi, F. G. (1953a). *Higher Transcendental Functions*. New York: McGraw-Hill.

Erdélyi, A., Magnus, W., Oberhettinger, F., & Tricomi, F. G. (1953b). *Higher Transcendental Functions Vol 2*. New York: McGraw-Hill.

Erdélyi, A., Magnus, W., Oberhettinger, F., & Tricomi, F. G. (1953c). *Tables of Integral Transforms Vol 2*. New York: McGraw-Hill.

Fano, A., Ritchie, D. W., & Carrieri, A. (2006). Modelling the structural basis of human CCR5 chemokine receptor function: from homology model-building and molecular dynamics validation to agonist and antagonist docking. *J. Chem. Inf. Model.* **46** (3), 1223–1235.

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters,* **7**, 861–874.

Fields, B. A., Malchiodi, E. L., Li, H., Ysern, X., Stauffacher, C. V., Schlievert, P. M., Karjalainen, K., & Mariuzza, R. A. (1996). Crystal structure of a T-cell receptor β-chain complexed with a superantigen. *Nature,* **384**, 188–192.

Frank, J. (2002a). Single-particle imaging of macromolecules by cryo-electron microscopy. *Ann. Rev. Biophys. Biolmol. Struct.* **31**, 303–319.

Frank, L. R. (2002b). Characterization of anisotropy in high angular resolution diffusion-weighted MRI. *Magnet. Reson. Med.* **47**, 1083–1099.

Funkhouser, T., Min, P., Kazhdan, M., Chen, D. Y., Halderman, A., & Dobkin, D. (2003). A search engine for 3D models. *ACM Transactions on Graphics,* **22**, 83–105.

Gabb, H. A., Jackson, R. M., & Sternberg, M. J. E. (1997). Modelling protein docking using shape complementarity, electrostatics and biochemical information. *J. Mol. Biol.* **272** (1), 106–120.

Ganser-Pornillos, B. K., Yeager, M., & Sundquist, W. I. (2008). The structural biology of hiv assembly. *Curr. Op. Struct. Biol.* **18**, 203–217.

Garboczi, E. (2002). Three-dimensional mathematical analysis of particle shape using X-ray tomography and spherical harmonics: Application to aggregates used in concrete. *Cement and Concrete Research,* **32**, 1621–1638.

Garzón, J. I., Lopéz-Blanco, J. R., Pons, C., Kovacs, J., Abagyan, R., & Chacón, P. (2009). FRODOCK: a new approach for fast rotational protein-protein docking. *Bioinformatics,* , Advanced Access, 20 July 2009: doi:10.1093/bioinformatics/btp447.

Gavin, A. C., Bosche, M., Krause, R., Grandi, P., Marzioch, M., Bauer, A., Schultz, J., Rick, J. M., Michon, A. M., & Cruciat, C. M. (2002). Functional organization of the yeast proteome by systematic analysis of protein complexes. *Nature,* **415**, 141 – 147.

Gillet, A., Sanner, M., Stoffler, D., & Olson, A. (2005). Tangible interfaces for structural molecular biology. *Structure,* **13**, 483–491.

Gnatt, A. L., Cramer, P., Fu, J., Bushnell, D. A., & Kornberg, R. D. (2001). Structural basis of transcription: An RNA polymerase II elongation complex at 3.3å resolution. *Science,* **292**, 1876–1882.

González-Ruiz, D. & Gohlke, H. (2006). Targeting protein-protein interactions with small molecules: challenges and perspectives for computational binding epitope detection and ligand finding. *Curr. Med. Chem.* **13**, 2607–2625.

Goodford, P. J. (1985). A computational procedure for determining energetically favourable binding sites on biologically important macromolecules. *J. Med. Chem.* **28**, 849–857.

Gottfried, K. (1966). *Quantum mechanics*. New York: Benjamin.

Grant, J. A., Gallardo, M. A., & Pickup, B. T. (1996). A fast method of molecular shape comparison: A simple application of a Gaussian description of molecular shape. *J. Comp. Chem.* **17** (14), 1653–1666.

Grant, J. A. & Pickup, B. T. (1995). A Gaussian description of molecular shape. *J. Phys. Chem.* **99**, 3503–3510.

Grigoriu, M., Garboczi, E., & Kafali, C. (2006). Spherical harmonic-based random fields for aggregates used in concrete. *Powder Technology,* **166**, 123–138.

Grünberg, R., Leckner, J., & Nilges, M. (2004). Complementarity of structure ensembles in protein-protein docking. *Structure,* **12**, 2125–2136.

Guéziec, A. & Hummel, R. (1995). Exploiting triangulated surface extraction using tetrahedral decomposition. *IEEE Trans. Vis. Comp. Graph.* **1** (4), 328–342.

Hart, G. T., Ramani, A. K., & Marcotte, E. M. (2006). How complete are current yeast and human protein interaction networks? *Genome Biol.* **7**, 120.

Hawkins, P. C. D., Skilllman, A. G., & Nicholls, A. (2007). Comparison of shape-matching and docking as virtual screening tools. *J. Med. Chem.* **50**, 74–82.

Helma, C., Cramer, T., Kramer, S., & De Raedt, L. (2004). Data mining and machine learning techniques for the identification of mutagenicity inducing substrates and structure activity relationships of noncongeneric compounds. *J. Chem. Inf. Comput. Sci.* **44**, 1402–1411.

Hinsen, K., Thomas, A., & Field, M. J. (1999). Analysis of domain motions in large proteins. *Proteins: Struct. Func. Genet.* **34**, 369–382.

Ho, Y., Gruhler, A., Heilbut, A., Bader, G. D., Moore, L., Adams, S., Millar, A., Taylor, P., Bennett, K., & Boutilier, K. (2002). Systematic identification of protein complexes in *saccharomyces cerevisiae* by mass spectrometry. *Nature,* **415**, 180 – 183.

Hobson, E. W. (1931). *The Theory of Spherical and Ellipsoidal Harmonics*. London: Cambridge University Press.

Hochstadt, H. (1971). *The Functions of Mathematical Physics*. New York: Wiley.

Holm, L. & Sander, C. (1991). Database algorithm for generating protein backbone and side-chain co-ordinates from a C_α trace. Application to model building and detection of co-ordinate errors. *J. Mol. Biol.* **218**, 183–194.

Huang, H., Shen, L., Zhang, R., Makedon, F., Hettleman, B., & Perlman, J. (2005). Surface alignment of 3D spherical harmonic models: Application to cardiac MRI analysis. *LNCS 3749 – Medical Image Computing and Computer-Assisted Intervention,* **8** (1), 67–74.

Huang, N. & Shoichet, B. K. (2006). Benchmarking sets for molecular docking. *J. Med. Chem.* **49** (23), 6789–6801.

Inbar, Y., Schneidman-Duhovny, D., Oron, A., Nussinov, R., & Wolfson, H. J. (2005). Approaching the CAPRI challenge with an efficient geometry-based docking. *Proteins: Struct. Func. Bioinf.* **60**, 217–223.

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

Ito, T., Chiba, T., Ozawa, R., Yoshida, M., Hattori, M., & Sakaki, Y. (2001). A comprehensive two-hybrid analysis to explore the yeast protein interactome. *Proc. Natl. Acad. Sci.* **98**, 4569–4574.

Jackson, J. D. (1975). *Classical Electrodynamics*. New York: Wiley.

Janin, J., Henrick, K., Moult, J., Ten Eyck, L., Sternberg, M. J. E., Vajda, S., Vakser, I., & Wodak, S. J. (2003). CAPRI: a critical assessment of predicted interactions. *Proteins: Struct. Func. Genet.* **52**, 2–9.

Jensen, F. (1999). *Introduction to Computational Chemistry*. New York: Wiley.

Jones, G., Willett, P., & Glen, R. C. (1995). Molecular recognition of receptor sites using a genetic algorithm with a description of desolvation. *J. Mol. Biol.* **245**, 43–53.

Kammler, D. (2000). *A First Course in Fourier Analysis*. Upper Saddle River, NJ: Prentice-Hall.

Katchalski-Katzir, E., Shariv, I., Eisenstein, M., Friesem, A. A., Aflalo, C., & Vakser, I. A. (1992). Molecular surface recognition: Determination of geometric fit between proteins and their ligands by correlation techniques. *Proc. Natl. Acad. Sci.* **89**, 2195–2199.

Kautz, J., Sloan, P. P., & Snyder, J. (2002). Fourier method for large-scale surface modeling and registration. *Proceedings of 13th Eurographics workshop on rendering,* **28**, 291–296.

Kazhdan, M., Funkhouser, T., & Rusinkiewcz, S. (2003). Rotation invariant spherical harmonic representation of 3D shape descriptors. *Proceedings of 2003 Eurographics/ACM SIGGRAPH symposium on geometry processing,* **43**, 156–164.

Keister, B. D. & Polyzou, W. N. (1997). Useful bases for problems in nuclear and particle physics. *J. Comp. Phys.* **134**, 231–235.

Kellenberger, E., Springael, J. Y., Parmentier, M., Hachet-Haas, M., Galzi, J. L., & Rognan, D. (2007). Identification of nonpeptide CCR5 receptor agonists by structure-based virtual screening. *J. Med. Chem.* **50**, 1294–1303.

Khodade, P., Prabhu, R., Chandra, N., Raha, S., & Govindrajan, R. (2007). Parallel implementation of autodock. *J. Appl. Cryst.* **40**, 598–599.

Kolodny, R., Koehl, P., & Levitt, M. (2005). Comprehensive evaluation of protein structure alignment methods: Scoring by geometric measures. *J. Mol. Biol.* **346**, 1173–1188.

Kovacs, J. A., Chacon, P., Cong, Y., Metwally, E., & Wriggers, W. (2003). Fast rotation matching of rigid bodies by fast Fourier transform acceleration of five degrees of freedom. *Acta Cryst.* **D59**, 1371–1376.

Kovacs, J. A. & Wriggers, W. (2002). Fast rotation matching. *Acta Cryst.* **D58**, 1282–1286.

Kozakov, D., Brenke, R., Comeau, S. R., & Vajda, S. (2006). PIPER: an FFT-based protein docking program with pairwise potentials. *Proteins: Struct. Func. Bioinf.* **65**, 392–406.

Kozakov, D., Clodfelter, K. H., Vajda, S., & Camacho, C. J. (2005). Optimal clustering for detecting near-native conformations in protein docking. *Biophys. J.* **89**, 867–875.

Lanczos, C. (1964). A precision approximation of the gamma function. *J. SIAM Numer. Anal.* **B1**, 86–96.

Lanzavecchia, S., Cantele, F., & Bellon, P. L. (2001). Alignment of 3D structures of macromolecular assemblies. *Bioinformatics,* **17** (1), 58–62.

Larson, R. S. (2006). *Bioinformatics and Drug Discovery*. New Jersey: Humana Press.

Lebedev, N. N. (1972). *Special Functions and Their Applications*. New York: Dover.

Lemmen, C. & Lengauer, T. (2000). Computational methods for the structural alignment of molecules. *J. Comput.-Aid. Mol. Des.* **14**, 215–232.

Lewis, P., De Jonge, M., Daeyaert, F., Koymans, L., Vinkers, M., Heeres, J., Janssen, P. A. J., Arnold, E., Das, K., Clark, A. D., Hughes, S. H., Boyer, P. L., De Béthune, M. P., Pauwels, R., Andries,

K., Kukla, M., Ludovici, D., De Corte, B., Cavas, R., & Ho, C. (2003). On the detection of multiple-binding modes of ligands to proteins, from biological, structural, and modeling data. *J. Comput.-Aid. Mol. Des.* **17**, 129–134.

Libbrecht, K. G. (1985). Practical considerations for the generation of large-order spherical harmonics. *Solar Physics,* **99** (1-2), 371–373.

Lin, J. & Clark, T. (2005). An analytical, variable resolution, complete description of static molecules and their intermolecular binding properties. *J. Chem. Inf. Model.* **45**, 1010–1016.

Lindert, S., Stewart, P. L., & Meiler, J. (2009). Hybrid approaches: applying computational methods in cryo-electron microscopy. *Curr. Op. Struct. Biol.* **19**, 218–225.

Luke, Y. L. (1969). *The Special Functions and their Approximation*. New York: Academic Press.

Macindoe, G., Mavridis, L., Venkatraman, V., Devignes, M.-D., & Ritchie, D. W. (2010). HexServer: an FFT-based protein docking server powered by graphics processors. *Nucleic Acids Res.* **38**, W445–W449.

Madej, T., Gibrat, J.-F., & Bryant, S. H. (1995). Threading a database of protein cores. *Proteins: Struct. Func. Bioinf.* **23** (3), 356–369.

Mak, L., Grandison, S., & Morris, R. J. (2008). An extension of spherical harmonics to region-based rotationally invariant descriptors for molecular shape description and comparison. *J. Mol. Graph. Model.* **26**, 1035–1045.

Mandell, J. G., Roberts, V. A., Pique, M. E., Kotlovyi, V., Mitchell, J. C., Nelson, E., Tsigelny, I., & Ten Eyck, L. F. (2001). Protein docking using continuum electrostatics and geometric fit. *Protein Eng.* **14** (2), 105–113.

Mathivavnan, S., Periaswamy, B., Gandhi, T. K. B., Kandasamy, K., Suresh, S., Mohmood, R., Ramachandra, Y. L., & Pandey, A. (2006). An evaluation of human protein-protein interaction data in the public domain. *BMC Bioinformatics,* **7**, S19.

Matter, H. & Schwab, W. (1999). Affinity and selectivity of matrix metalloproteinase inhibitors: a chemometrical study from the perspective of ligands and proteins. *J. Med. Chem.* **42**, 4506–4523.

Mavridis, L., Hudson, B. D., & Ritchie, D. W. (2007). Toward high throughput screening using spherical harmonic surface representations. *J. Chem. Inf. Model.* **47** (5), 1878–1796.

Mavridis, L. & Ritchie, D. W. (2009). 3D-Blast: protein protein structure alignment, comparison, and classification using spherical polar fourier correlations. *Pacific Symposium on Biocomputing,* **2010**, 281–292.

May, A. & Zacharias, M. (2008). Energy minimization in low-frequency normal modes to efficiently allow for global flexibility during systematic protein-protein docking. *Proteins: Struct. Func. Bioinf.* **70**, 794–809.

McPeek, M. A., Shen, L., & Farid, H. (2009). The correlated evolution of three-dimensional reproductive structures between male and female damsel flies. *Evolution,* **63**, 73–83.

McPeek, M. A., Shen, L., & Torrey, J. Z. (2008). The tempo and mode of three-dimensional morphological evolution in male reproductive structures. *Americal Naturalist,* **171**, E158–E178.

Méndez, R., Leplae, R., De Maria, L., & Wodak, S. J. (2003). Assessment of blind predictions of protein-protein interactions: current status of docking methods. *Proteins: Struct. Func. Genet.* **52**, 51–67.

Méndez, R. & Wodak, S. J. (2005). Assessment of CAPRI predictions in rounds 3-5 shows progress in docking procedures. *Proteins: Struct. Func. Bioinf.* **60**, 150–169.

Mintseris, J., Wiehe, K., Pierce, B., Anderson, R., Chen, R., Janin, J., & Weng, Z. (2005). Protein-protein docking benchmark 2.0: An update. *Proteins: Struct. Func. Bioinf.* **60**, 214–216.

Morris, R. J., Najmanovich, R. J., Kahraman, A., & Thornton, J. (2005). Real spherical harmonic expansion coefficients as 3D shape descriptors for protein binding pocket and ligand comparisons. *Bioinformatics,* **21**, 2347–2355.

Muench, S. P., Huss, M., Song, C. F., Phillips, C., Wieczorek, H., Trinick, J., & Harrison, M. A. (2009). Cryo-electron microscopy of the vacuolar ATPase motor reveals its mechanical and regulatory complexity. *J. Mol. Biol.* **386**, 989–999.

Muggleton, S. (2005). Machine learning for systems biology. In: *15th International Conference on Inductive Logic Programming*, (Kramer, S. & Pfahringer, B., eds) pp. 416–423, Bonn: Springer LNAI 3625.

Mustard, D. & Ritchie, D. W. (2005). Docking essential dynamics eigenstructures. *Proteins: Struct. Func. Bioinf.* **60**, 269–274.

Nagata, A., Mizushima, H., & Tanaka, H. (2002). Concept and prototype of protein-ligand docking simulator with force feedback technology. *Bioinformatics,* **18**, 140–146.

Napoli, A. (2005). A smooth introduction to symbolic methods for knowledge discovery. In: *Handbook of Categorization in Cognitive Science*, (Cohen, H. & Lefebvre, C., eds) pp. 813–933, Amsterdam: Elsevier.

Nilges, M. (1995). Calculation of protein structures with ambiguous distance restraints. automated assignment of ambiguous NOE crosspeaks and disulphide connectivities. *J. Mol. Biol.* **245**, 645–660.

Novotni, M. & Klein, R. (2003). 3d zernike descriptors for content based shape retrieval. *Proceedings of the eighth ACM symposium on Solid modeling and applications,* **SPM08**, 216–225.

Orengo, C. A., Michine, A. D., Jones, S., Jones, D. T., Swindells, M. B., & Thornton, J. M. (1997). CATH - A hierarchic classification of protein domain structures. *Structure,* **5** (8), 1093–1108.

Owens, J. D., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A. E., & Purcell, T. J. (2007). A survey of general-purpose computation on graphics hardware. *Comp. Graph. Forum,* **26**, 80–113.

Papageorgiou, A. C., Collins, C. M., Gutman, D. M., Kline, J. B., O'Brien, S. M., Tranter, H. S., & Acharya, K. R. (1995). Structural basis for the recognition of superantigen streptococcal pyrogenic exotoxin A (SpeA1) by MHC class II molecules and T-cell receptors. *EMBO J.* **18**, 9–21.

Park, H., Lee, J., & Lee, S. (2006). Critical assessment of the automated autodock as a new docking tool for virtual screening. *Proteins: Struct. Func. Genet.* **65**, 594–554.

Pastor, M. & Cruciani, G. (1995). A novel strategy for improving ligand selectivity in receptor-based drug design. *J. Med. Chem.* **38**, 4637–4647.

Pérez-Nueno, V. I., Ritchie, D. W., Rabal, O., Pascual, R., Borrell, J. I., & Teixidó, J. (2008). Comparison of ligand-based and receptor-based virtual screening of HIV entry inhibitors for the CXCR4 and CCR5 receptors using 3D ligand shape matching and ligand-receptor docking. *J. Chem. Inf. Model.* **48** (3), 509–533.

Petsko, G. A. & Ringe, D. (2004). *Protein Structure and Function*. London: New Science Press.

Pevsner, J. (2003). *Bioinformatics and Functional Genomics*. New York: Wiley.

Phuong, T. & Ho, T. B. (2005). Prediction of domain-domain interactions using inductive logic programming from multiple genome databases. In: *Ninth International Conference on Discovery Science*, (Lavrac, N., Todorovski, L., Jantke, K., & Klaus, P., eds) pp. 185–196, Bonn: Springer LNAI 4256.

Richards, W. G. (2002). Virtual screening using grid computing: the screensaver project. *Nature Rev. Drug. Disc.* **1**, 551–555.

Richmond, T. J. (1984). Solvent accessible surface area and excluded volume in proteins. *J. Mol. Biol.* **178**, 63–89.

Ritchie, D. W. (1998). *Parametric Protein Shape Recognition*. PhD thesis University of Aberdeen U.K.

Ritchie, D. W. (2003). Evaluation of protein docking predictions using *Hex* 3.1 in CAPRI rounds 1 and 2. *Proteins: Struct. Func. Genet.* **52** (1), 98–106.

Ritchie, D. W. (2005). High-order analytic translation matrix elements for real-space six-dimensional polar Fourier correlations. *J. Appl. Cryst.* **38**, 808–818.

Ritchie, D. W. (2008). Recent progress and future directions in protein-protein docking. *Curr. Prot. Pep. Sci.* **9** (1), 1–15.

Ritchie, D. W. & Kemp, G. J. L. (1999). Fast computation, rotation and comparison of low resolution spherical harmonic molecular surfaces. *J. Comp. Chem.* **20** (4), 383–395.

Ritchie, D. W. & Kemp, G. J. L. (2000). Protein docking using spherical polar Fourier correlations. *Proteins: Struct. Func. Genet.* **39** (2), 178–194.

Ritchie, D. W., Kozakov, D., & Vajda, S. (2008). Accelerating protein-protein docking correlations using a six-dimensional analytic FFT generating function. *Bioinformatics,* **24** (4), 810–823.

Ritchie, D. W. & Venkatraman, V. (2010). Ultra-fast FFT protein docking on graphics processors. *Bioinformatics,* **26**, 2398–2405.

Robinson, R. C., Turbedsky, K., Kaiser, D. A., Marchand, J. B., Higgs, H. N., Choe, S., & Pollard, T. D. (2001). Crystal structure of arp2/3 complex. *Science,* **294**, 1679–1684.

Rose, M. E. (1957). *Elementary Theory of Angular Momentum*. New York: Wiley.

Roseman, A. M. (2000). Docking structures of domains into maps from cryo-electron microscopy using local correlation. *Acta Cryst.* **D56**, 1332–1340.

Rossmann, M. G. (2000). Fitting atomic models into electron-microscopy maps. *Acta Cryst.* **D56**, 1341–1349.

Rossmann, M. G., Arisaka, F., Battisti, A. J., Bowman, V. D., Chipman, P. R., Fokine, A., Halfstein, S., Kanamura, S., Kostyuchemko, V. A., Mesyanzhinov, V. V., Schneider, M. M., Morais, M. C., Leiman, P. G., Palermo, L. M., Parrish, C. R., & Xiao, C. (2007). From structure of the complex to understanding of the biology. *Acta Cryst.* **D63**, 9–16.

Rost, B. (1999). Twilight zone of protein sequence alignments. *Protein Eng.* **12**, 85–94.

Rueda, M., Bottegoni, G., & Abagyan, R. (2009). Consistent improvement of cross-docking results using binding site ensembles generated with elastic network normal modes. *J. Chem. Inf. Model.* **49**, 716–725.

Russell, R. B., Alber, F., Aloy, P., Davis, F. P., Korkin, D., Pichaud, M., Topf, M., & Sali, A. (2004). A structural perspective on protein-protein interactions. *Curr. Op. Struct. Biol.* **14**, 313–324.

Sael, L., , La, D., Fang, Y., Ramani, K., R.Rustamov, & Kihara, D. (2008). Fast protein tertiary structure retrieval based on global surface shape similarity. *Proteins: Struct. Func. Bioinf.* **72**, 1259–1273.

Sakurai, J. J. (1994). *Modern Quantum Mechanics*. Reading, MA: Addison-Wesley.

Shen, L., Farid, H., & McPeek, M. A. (2009a). Modeling three-dimensional morphological structures using spherical harmonics. *Evolution,* **63**, 1003–1016.

Shen, L., Kim, S., & Saykin, A. J. (2009b). Fourier method for large-scale surface modeling and registration. *Computers & Graphics,* **33**, 299–311.

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

Sippl, M. J. & Wiederstein, M. (2008). A note on difficult structure alignment problems. *Bioinformatics,* **24**, 426–427.

Smith, G. R., Sternberg, M. J. E., & Bates, P. A. (2005). The relationship between the flexibility of proteins and their conformational states on forming protein-protein complexes with an application to protein-protein docking. *J. Mol. Biol.* **347**, 1077–1101.

Srinivasan, K., Mahawar, H., & Sarin, V. (2005). A multipole based treecode using spherical harmonics for potentials of the form $r^{-\lambda}$. *Lect. Notes Comp. Sci.* **3514**, 107–104.

Stein, M., Gabdoulline, R. R., & Wade, R. C. (2007). Bridging from molecular simulations to biochemical networks. *Curr. Op. Struct. Biol.* **17**, 166–172.

Sticht, J., Humbert, M., Findlow, S., Bodem, J., Müller, B., Deitrich, U., Werner, J., & Kr"ausslich, H. G. (2005). A peptide inhibitor of HIV-1 assembly *in vitro*. *Nature Struct. Biol.* **12** (8), 671–677.

Stone, J. E., Phillips, J. C., Freddolino, P. L., Hardy, D. J., Trabuco, L. G., & Schulten, K. (2007). Accelerating molecular modeling applications with graphics processors. *J. Comp. Chem.* **28**, 2618–2640.

Stowell, M. H. B., Miyazawa, A., & Unwin, N. (1998). Macromolecular structure determination by electron microscopy: new advances and recent results. *Curr. Op. Struct. Biol.* **8**, 606–611.

Sussman, J. L., Lin, D., Jiang, J., Manning, N. O., Prilusky, J., Ritter, O., & Abola., E. E. (1998). Protein data bank (PDB): database of three-dimensional structural information of biological macromolecules. *Acta Cryst.* **D54**, 1078–1084.

Taha, M. O., Qandil, A. M., Zaki, D. D., & Aldaben, M. A. (2005). Ligand-based assessment of factor Xa binding site flexibility via elaborate pharmacophore exploration and genetic algorithm-based QSAR modeling. *Eur. J. Med. Chem.* **40**, 701–727.

Takana, S. Y. & Mitchell, J. B. O. (2004). A structure-odour relationship study using EVA descriptors and hierarchical clustering. *Organic Biomol. Chem.* **22** (2), 3250–3255.

Talman, J. D. (1968). *Special Functions: A Group Theoretical Approach*. New York: W. A. Benjamin Inc.

Tama, F., Gadea, F. X., Marques, O., & Sanejouand, Y. (2000). Building-block approach for determining low-frequency normal modes of macromolecules. *Proteins: Struct. Func. Genet.* **41**, 1–7.

Tama, F. & Sanejouand, Y. H. (2001). Conformational change of proteins arising from normal mode calculations. *Protein Eng.* **14** (1), 1–6.

Taylor, W. R. (1999). Protein structure comparison using iterated double dynamic programming. *Protein Sci.* **8**, 654–665.

Taylor, W. R. & Orengo, C. A. (1989). Protein structure alignment. *J. Mol. Biol.* **208**, 1–22.

Tirion, M. (1996). Large amplitude elastic motions in proteins from a single-parameter, atomic analysis. *Phys. Rev. Lett.* **77**, 1905–1908.

Uetz, P., Giot, L., Cagney, G., Mansfield, T. A., Judson, R. S., Knight, J. R., Lockshon, D., Narayan, V., Srinivasan, M., Pochart, P., Qureshi-Emili, A., Li, Y., Godwin, B., Conover, D., Kalbfleisch, T., Vijayadamodar, G., Yang, M. J., Johnston, M., Fields, S., & Rothberg, J. M. (2000). A comprehensive analysis of protein-protein interactions in *saccaromyces cerevisiae*. *Nature,* **403**, 623–671.

Vakser, I. A. & Aflalo, C. (1994). Hydrophobic docking: A proposed enhancement to molecular recognition techniques. *Proteins: Struct. Func. Genet.* **20**, 320–329.

Venkatraman, V., Sael, L., & Kihara, D. (2009). Potential for protein surface shape analysis using spherical harmonics and 3D Zernike descriptors. *Cell Biochem. Biophys,* **54**, 23–32.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association,* **58**, 236–244.

Weiner, S. J., Kollman, P. A., Case, D. A., Singh, U. C., Ghio, C., Alagona, G., Profeta Jr., S., & Weiner, P. (1984). A new force field for molecular mechanical simulation of nucleic acids and proteins. *J. Am. Chem. Soc.* **106**, 765–784.

Weniger, E. J. & Steinborn, E. O. (1983). The Fourier transforms of some exponential-type basis functions and their relevance to multicenter problems. *J. Chem. Phys.* **78**, 6121–6132.

Wiggins, R. A. & Saito, M. (1971). Evaluation of computational algorithms for the associated Legendre polynomials by interval analysis. *Bull Seismol. Soc. Am.* **61** (2), 375–381.

Wigner, E. P. (1939). On the unitary representations of the inhomogeneous repesentation of the Lorentz group. *Annals of Mathematics,* **40** (1), 149–204.

Williams, S., Bledsoe, R. K., Collins, J. L., Boggs, S., Lambert, M. H., Miller, A. B., Moore, J., McKee, D. D., Moore, L., Nichols, J., Parks, D., Watson, M., Wisely, B., & Willson, T. M. (2003). X-ray crystal structure of the liver X receptor β ligand binding domain: Regulation by a histidine-tryptophan switch. *J. Biol. Chem.* **278**, 27138–27143.

Wollacott, A. M. & Mertz, K. M. (2005). Haptic applications for molecular structure manipulation. *J. Mol. Graph. Model.* **25**, 801–805.

Wong, R. S., Bodart, V., Metz, M., Labrecque, J., Bridger, G., & Fricker, S. P. (2006). Prediction of multiple binding modes of the CDK2 inhibitors, anilinopyrazoles, using the automated docking programs GOLD, FlexX, and LigandFit: an evaluation of performance. *J. Chem. Inf. Model.* **46**, 2552–2562.

Wong, R. S., Bodart, V., Metz, M., Labrecque, J., Bridger, G., & Fricker, S. P. (2008). Comparison of the potential multiple binding modes of bicyclam, monocylam, and noncyclam small-molecule CXC chemokine receptor 4 inhibitors. *Mol. Pharmacol.* **74**, 1485–1495.

Wriggers, W., Milligan, R. A., & McCammon, J. A. (1999). Situs: A package for docking crystal structures into low-resolution maps from electron density. *J. Struct. Biol.* **125**, 185–195.

Yamagishi, M. E. B., Martins, N. F., Neshich, G., Cai, W., Shao, X., Beautrait, A., & Maigret, B. (2006). A fast surface-matching procedure for protein-ligand docking. *J. Mol. Model.* **12** (2), 965–972.

Appendix A

Relevant Publications

This Appendix contains copies of the following peer-reviewed journal articles:

- Fast computation, rotation, and comparison of low resolution spherical harmonic molecular surfaces. D.W. **Ritchie**, G.J.L. Kemp. *J. Comp. Chem.* **20**(4), 383–395 (1999).
- Protein docking using spherical polar Fourier correlations. D.W. **Ritchie**, G.J.L. Kemp. *Proteins: Struct. Func. Genet.* **39**, 178–194 (2000).
- Evaluation of protein docking predictions using Hex 3.1 in CAPRI rounds 1 and 2. D.W. **Ritchie**. *Proteins: Struct. Func. Genet.* **52**, 98–106 (2003).
- Docking essential dynamics eigenstructures. D. Mustard and D.W. **Ritchie**. *Proteins: Struct. Funct. Bioinf.* **60**, 269–274 (2005).
- High order analytic translation matrix elements for real space six-dimensional polar Fourier correlations. D.W. **Ritchie**. *J. Appl. Cryst.* **38** 808–818 (2005).
- Modelling the structural basis of human CCR5 chemokine receptor function: from homology model-building and molecular dynamics validation to agonist and antagonist docking. A. Fano, D.W. **Ritchie**, A. Carrieri. *J. Chem. Inf. Model.* **46**(3) 1223–1235 (2006).
- Toward high throughput screening using spherical harmonic surface representations. L. Mavridis, B.D. Hudson, D.W. **Ritchie**. *J. Chem. Inf. Model.* **47**(5) 1787–1796 (2007).
- Comparison of ligand-based and receptor-based virtual screening of HIV entry inhibitors for the CXCR4 and CCR5 receptors using 3D ligand shape matching and ligand-receptor docking. V.I. Pérez-Nueno, D.W. **Ritchie**, O. Rabal, R. Pascual, J.I. Borel, J. Teixidó. *J. Chem. Inf. Model.* **48**(3) 509–533 (2008).

- Recent progress and future directions in protein-protein docking. D.W. **Ritchie**. *Curr. Prot. Pep. Sci.* **9**(1) 1–15 (2008).
- Accelerating protein-protein docking correlations using a six-dimensional analytic FFT generating function. D.W. **Ritchie**, D. Kozakov, and S. Vajda. *Bioinformatics* **24**(4) 810–823 (2008).
- Clustering and Classifying Diverse HIV Entry Inhibitors Using a Novel Consensus Shape-Based Virtual Screening Approach: Further Evidence for Multiple Binding Sites within the CCR5 Extracellular Pocket. V.I. Pérez-Nueno, D.W. **Ritchie**, J.I. Borrell, and J. Teixidó. *J. Chem. Inf. Model.* **48**(11) 2146–2165 (2008).
- 3D-Blast: 3D protein structure alignment, comparison, and classification using spherical polar Fourier correlations. L. Mavridis and D.W. **Ritchie**. *Pacific Symposium on Biocomputing (PSB 2010),* 281–292.
- SHREC-10 Track: Protein Models. L. Mavridis, V. Venkatraman, D. W. Ritchie, N. Morikawa, R. Andonov, A. Cornu, N. Malod-Dognin, J. Nicolas, M. Temerinac-Ott, M. Reisert, H. Burkhardt, A. Axenopoulos (2010). 3DOR: Eurographics Workshop on 3D Object Retrieval (2010), 117–124.
- Ultra-Fast Protein Docking on Graphics Processors. D.W. **Ritchie**, V. Venkatraman, (2010). *Bioinformatics*. **26**, 2398–2405.
- HexServer: an FFT-based protein docking server powered by graphics processors. G. Macindoe, L. Mavridis, V. Venkatraman, M.-D. Devignes, D.W. **Ritchie** (2010). *Nucleic Acids Research*, **38**, W445–W449.