An Adaptive Irregularly Spaced Fourier Method for Protein-Protein Docking *

Julio E. Castrillón-Candás † Chandrajit Bajaj ‡ Vinay Siddavanahalli §

Department of Computer Sciences, & Institute of Computational Engineering and Sciences,
Computational Visualization Center
University of Texas at Austin
Austin, TX 78712

July 11, 2005

Abstract

In this paper we introduce a grid free irregularly sampled Fourier approach for accurately predicting rigid body protein-protein docking sites. Of the many docking approaches, grid based Fast Fourier Transform (FFT) approaches have been shown to produce by far the fastest, correlation profiles of complex protein-protein interactions over the six dimensional search space. However, these uniform sampling methods still possess high time complexity, and in particular, are highly memory intensive for predicting large protein-protein docking sites. By taking advantage of an irregularly and adaptively sampled, smooth particle representation of molecular shape, in combination with the use of irregularly spaced FFT transforms, we eliminate an explicit uniform grid. We are able to produce efficiently, highly compressed, but accurate, docking correlation profiles.

*This work was supported in part by NSF grants ACI-0220037, EIA-0325550, and NIH grants 0P20 RR020647, and R01 GM074258
†julio@ices.utexas.edu
‡bajaj@cs.utexas.edu
§skvinay@cs.utexas.edu

1 Introduction

Efforts in structural proteomics have lead to a rapid increase in the number of three-dimensional (3-D) structures of individual proteins. Moreover, knowledge of networks of interactions and signaling pathways is also expanding rapidly through genomic and proteomics approaches. Still, our picture of the structures of both stable and transient protein interactions lags behind. Efforts in crystallizing macromolecular complexes have met with limited success, and hybrid experimental approaches, utilizing cryo-electron microscopy and crystallography or NMR to give structural details of complex assemblies are evolving. However, along with these experimental methods, there is a growing need for efficient and robust computational approaches to predicting the complexed viable structures in protein-protein interactions. These approaches are also known as protein-protein docking.

Protein-protein docking or in general molecular docking usually consists of two primary selections. One is the choice of goodness of fit measure (sometimes called the scoring function) while the other is the choice of the search algorithm. Both of these decisions are based on an assumed molecular model. The scoring function includes consideration for molecular properties in addition to a representation of molecular shape. Grid based Fast Fourier Transform (FFT) approaches have been shown to produce highly accurate correlation profiles of complex protein-protein docking making them a popular choice for solving the above docking site search problem. However, they are time consuming, and in particular, highly memory intensive for large molecules due to the large size of the grid needed. In this paper we introduce an adaptive grid-free irregularly spaced Fourier approach for accurately predicting rigid body protein-protein docking sites.

Problem Description

For molecule A, let $V_i^A : \mathbb{R}^3 \rightarrow \mathbb{R}$ be the i^{th} associated density map for $i = 1 \dots m$, where each map represents a molecular shape or property. Similarly for molecule B we have $V_i^B : \mathbb{R}^3 \rightarrow \mathbb{R}$ maps for $i = 1 \dots m$. Let $S_i(V_i) : \mathbb{R}^3 \rightarrow \mathbb{C}$ be the scoring function defined on V_i. For a rotation R in the 3D rotation group $SO(3)$, the rotation operator Λ_R is defined as

$$\Lambda_R S(\vec{x}) := S(R^{-1}(\vec{x})) \quad \forall \vec{x} \in \mathbb{R}^3,$$

where $\vec{x} = (x, y, z)$. Similarly, the translator operator $T^{j,k,l}$ is defined as

$$T^{j,k,l}(S_i(x, y, z)) = S_i(x - j, y - k, z - l)$$

for $j, k, l \in \mathbb{R}$. The six dimensional search docking problem, can be posed as

$$\arg\max_{j,k,l,R} \sum_{i=1}^{m} v_i \int_{\vec{x}\in\mathbb{R}^3} \int_{\vec{y}\in\mathbb{R}^3} \Lambda_R(S_i(V_i^A(\vec{x})))T^{j,k,l}(V_i^B(\vec{y}))\ d\vec{x}d\vec{y},$$

where v_i are calibration weights. Note that alternate formulations lead to the same search space. Indeed, in [34] split the search problem into 5D rotations and a 1D translation.

Molecular shapes have a natural smooth particle atomistic or quasi-atomistic representation. By taking advantage of the adaptive smooth particle representation, we eliminate the underlying grid thus producing highly

compressed, but accurate, correlation profiles based on an adaptive irregularly spaced FFT algorithm. Our docking method primarily consists of three steps: First, we select an adaptive smooth particle representation for proteins which is also compatible with our initial shape-complementarity based scoring function. Second, we calculate the frequency profiles directly from the smooth particle representation, and search effectively over six dimensional translation and rotational space, utilizing the irregularly spaced FFT, and finally, we evaluate a compressed correlation profile which captures the rigid body protein-protein docking sites.

The rest of the paper is as follows. In section 2, we summarize the main Fourier based approaches to the rigid body protein-protein docking problem. Moreover, the different approaches to the FFT over irregularly sampled domains are described. A complexity analysis of grid and spherical harmonic Fourier based algorithms for docking and matching is given in Appendix A. In section 3 the main part of the algorithm is described. In section 3.1 a smooth particle representation of molecular maps and affinity functions is introduced alongwith the corresponding shape complementarity based scoring function to capture rigid body protein-protein docking.

With a suitable shape complementarity based scoring function defined, the search algorithm is separated into two parts: the translational Fourier based search and the rotational search. In section 3.3 we show how to reduce the computational and storage costs of the translation search algorithm with our method. Traditional grid based Fourier approach embeds the two molecular maps in a N^3 grid and convolve them using the FFT leading to $\mathcal{O}(N^3 \log N)$ time. In our irregularly spaced Fourier method, we assume both molecules to contain M atoms (you can take the maximum number of atoms from both molecules). An accurate approximate correlation profile is derived in $\mathcal{O}(M \log M)$ computational steps and $\mathcal{O}(M)$ storage. In practice M is much smaller that N^3.

Finally, in section 4 we describe our implementation and report on a few docking results including the actual timing and the accuracy of our correlation profiles.

2 Prior Work

In this section we briefly review past docking approaches with an emphasis on techniques applying Fourier search. A review on irregularly sampled Fourier transforms in also presented.

2.1 Molecular Shape and Affinity Functions

Various molecular surfaces have been defined (Figure 1) using a spherical representation of individual atoms and a spherical probe representing a solvent molecule. The SAS is outlined by the center of the probe sphere as it "rolls" over the atoms constituting a molecule [11]. The SES [25], [11], is defined as the inner boundary of the volume that can be occupied by solvent in contact with the molecule. A number of algorithms have been developed to compute these surfaces [1, 3, 4, 11, 29, 35, 36, 40–42] for the purpose of visualization and various computations. It is interesting to observe that the SES of proteins forming molecular complexes exhibits a very high level of geometrical complementarity. These surfaces are used extensively for visualizing and studying molecular properties and interactions. However, these surfaces are approximations of a somewhat fuzzy boundary of the molecule's electron density. Surfaces are also used to visualize molecular properties associated with molecular shape (e.g. charge density, electrostatic potential, hydrophobicity, *etc*.)([12, 22]). Such surfaces are usually level sets of scalar fields and their gradient or Laplacian ([11, 29, 36, 41, 42]).

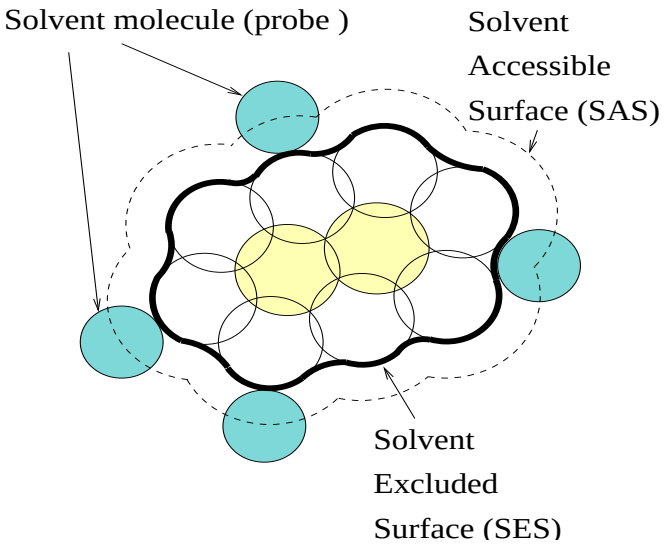


Figure 1: Solvent Accessible and Solvent Excluded Surfaces.

Molecular shape (surface and volumetric) are also derived from approximations of an appropriate level set of electron density [6, 13, 28, 33]. The accurate computation of electron density representations for molecules from the PDB requires computations at the quantum mechanical level [7]. One usually approximates the electron density distribution of the i^{th} atom with a Gaussian function ([3, 6, 7, 20, 30, 31, 37]) as

$$\rho_i(\mathbf{r}) = \exp\left(\frac{\mathcal{B}r^2}{R_i^2} - \mathcal{B}\right),$$

where $\mathcal{B} < 0$ is the rate of decay parameter, R_i is the Van der Waals radius of the i^{th} atom and $r^2 = (x - x_{ci})^2 + (y - y_{ci})^2 + (z - z_{ci})^2$ ($\{x_{ci}, y_{ci}, z_{ci}\}$ is the center of the i^{th} atom). A volumetric representation of the molecule may now be obtained by summing the contributions from each single atom, thus the electron density $I(\vec{x})$ for M atoms is described as

$$I(\vec{x}) = \sum_{k=1}^{M} \rho_k(r) = \sum_{k=1}^{M} e^{(\frac{\mathcal{B}r^2}{R_k^2} - \mathcal{B})}. \quad (1)$$

Notice that for protein structures, R_k can be grouped into a set of about 15 distinct values.

A critical component of all docking approaches is defining a suitable measure for the affinity functions in the scoring calculations. Paper [23] separates the affinity functions into core and a surface skin with the objective to penalize core-core clashes, but add positively surface skin-surface skin overlaps. By assigning different affinities to the core and the molecular surface skin of each atom and performing a convolution between these weighted maps, a profile is obtained where the largest values conform to the best translational overlap. Modifications of this approach have been developed in ([9,10,19,26]). They define the core and the skin regions using

the molecular surfaces like the *solvent accessible surface* (SAS) and *solvent excluded surface* (SES). Other approaches include adapting scoring functions for molecular *matching* [8, 21]. These scoring functions are also designed to match molecular functional properties, such as electrostatics potential. They can be modified for docking by forming a function f for molecule A and g for the complementary volume for molecule B.

2.2 Grid Based Fourier Methods

Katchalski-Katzir et. al.'s [23] use coarse grids and rotational angles to reduce the combinatorics of the search. Gabb et. *al.* [19] use the a priori knowledge of suitable binding site locations on the proteins to reduce the combinatorics of possible relative conformations. Fast Fourier Transforms are used in each of [19, 23, 34] to additionally speed up the cumulative scoring function computations and hence the search. Moreover, in [9, 10] Chen et. *al.* improve on FFT Grid based methods [19] with better scoring functions and additional molecular properties.

2.3 Spherical Harmonic Fourier Methods

Several groups [13, 28, 34] studied the problem of representing molecular surfaces with expansions of spherical harmonic functions and its application to fast computations of the protein docking problem.

Efficiency is additionally gained from the fast rotation and cumulative correlation function computations involving coefficients of spherical harmonic polynomials. To combat the numerical intensive trigonometric computations in these methods, many values are precomputed and cached in a direct trade-off of memory for increased speed. For example, most of the sine and cosine terms of the spherical harmonic expansion are cached [33]. Additionally, pre-calculated values of the functions K(R) of the intermolecular separation R, are stored. Ritchie et. al.'s [33] results compare favorably to those of the geometry method [17] and the Cartesian FFT correlation method of Gabb et. al. [19].

However, the approach of [33] has some limitations. Their choice of parametric spherical harmonic expansions in is valid only for molecules exhibiting spherical topology. Secondly, a fixed and by no means small truncation order of spherical harmonic polynomials (typically 25 orders or more) is required to approximate the density/characteristic functions. The non-adaptivity of the surface approximation based on a single point spherical expansion, also makes it difficult to directly relate the expansion order to the range of errors that the approximation generates. Third, the storage required to cache K(R) is not inconsequential. For example, in [33], values of K(R) in 1 (Å) increments are stored using 55MB of disk space. Finally, uniform icosahedral sampling, used by Ritchie for discretization of the scoring function (reducing the integrals to discrete sums) and the Fourier calculation, is not guaranteed to be accurate for proteins that have rotationally skewed aspect ratios (i.e. elongated along a single dimension). In [24], a similar Fourier method for fast rotational matching using spherical harmonics, overcomes the restriction to star shaped molecules by discretizing the volumetric space into several shells. The price to pay, is the prohibitively expensive memory usage.

In the appendix A, a more accurate description of spherical harmonics methods [24, 33] is given. Moreover, a complexity analysis between [24, 33], grid based methods (Zdock [9, 10]) and our adaptive irregularly spaced Fourier is presented.

2.4 Irregularly Sampled Fourier Transforms

Several approaches to efficiently compute the Discrete Fourier Transform (DFT) polynomial for irregularly sampled domains have surfaced during the last decade [2, 5, 14–16, 32, 38, 39]. A review of many of these approaches can be found in [43]. In [39] the domain is split into subintervals and each subinterval is then projected onto a space of local Chebyshev polynomials. An alternate expansion is done in [2], Chebyshev polynomials are replaced with a Taylor expansion. For Fourier Transforms with singularities, Beylkin employs a series projections onto multi-resolution spaces [5]. Dutt et. *al* ([14, 15]) represent the DFT polynomial as a multi-pole expansion. For M non equidistant samples, the multi-pole approximate construction obtains the first M frequencies in $\mathcal{O}(M \log M)$ computational steps and $O(M)$ storage. The drawbacks are that such constructions require a fast multi-pole method, which leads to a complex implementation. Many of these approaches have been introduced for 1D domains. Extensions to multi-dimensions are possible through tensor products. In this paper, we decided to follow the Nonequidistant Fast Fourier Transform (NFFT) and NFFT′ approach of Potts, Elbel and Steidl ([16, 32, 38]), since 3D results were explicitly shown in their papers. Moreover, the NFFT is highly accurate, conceptually simple, and easy to implement.

3 Docking Algorithm

In this section we give a description of our approach for the fast search and scoring during the docking of two molecules A & B. The first step is to define molecular shape, affinity functions and population algorithm of the SAS skin. The scoring function provides a description of molecular interactions. The second step is developing a fast docking search algorithm.

3.1 Shape Complementarity

For the representation of shape we will define two regions for each molecule, an interior region called the "core" and a boundary region called the "skin". In a docking calculation one molecule is held fixed (termed receptor and designated molecule A here) while the other molecule (termed ligand and designated molecule B here) performs the 6-D search. The skin and core regions will be defined differently for molecule A and B, but all these regions will be represented using Gaussian expansions. Molecular affinity functions defined over these regions will be used to compute a complementarity score for any configuration of B around A. This score will be the combination over multiple properties. For each property, the complementarity score will be computed as a function of the overlap of the core and skin regions of the two molecules.

Figure 2(a) shows how we define skin and core regions for molecule A and B. Note that this definition is asymmetric as the skin of molecule A is completely outside molecule A, while the skin of B is defined using surface atoms of B and hence is completely inside molecule B.

The following types of overlap will be possible: 1) CoreA-CoreB and coreA-skinB overlaps are steric clashes; 2) SkinA-skinB are the most favorable overlaps; 3) SkinA-coreB correspond to volumes outside molecule A overlapping with interior atoms of B. This term will be ignored as it neither penalizes nor contributes to the quality of the fit. In Figure 2(a) we show how the two molecules (A) and (B) dock.

To define the skin-layer and the interior we need to obtain centers suitable for a smooth particle representation of these two regions. In appendix B our population algorithm computes the skin layers for molecules A

and B. Examples of computed skin layers are shown in Figure 3. In (a-c) the skin-layer and core regions for molecule A is shown. In Figure (d-f) we have an example of the skin layer for molecules B.

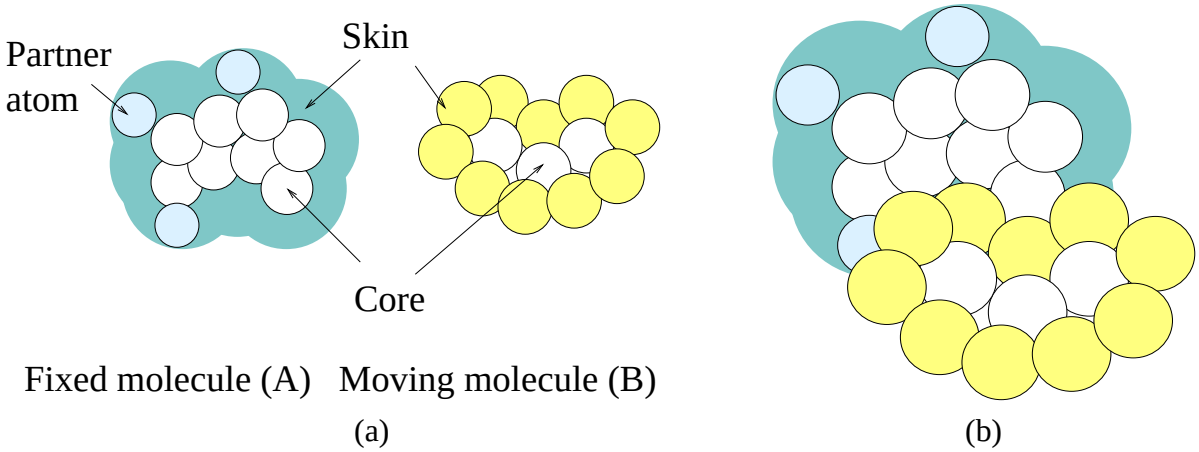


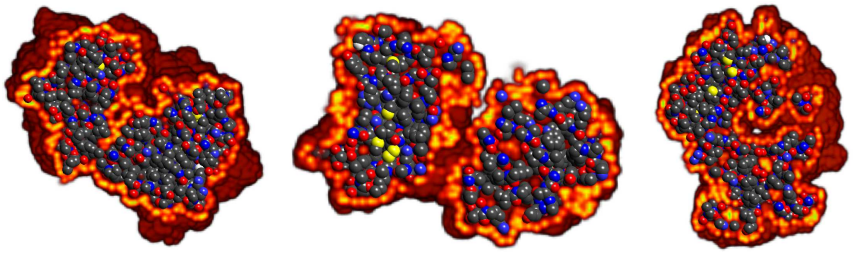
Figure 2: (a) Skin and Core regions. Atoms are drawn as solid circles. The skins regions are colored while the core regions and white. (b) Docking of molecules (A) & (B).

The regions shown in Figure 2 define domains over which various affinity functions can be represented using a smooth particle representation. For instance, the shape property can be represented by placing a Gaussian on every center of the region.

We can also define a finite number of groups of centers in the smooth particle representations and assign a particular weight to each group. This weight will scale the contribution of (the function placed on) the centers belonging to the group to the final score. This mechanism will allow the representation of water molecules for instance, by adding oxygen atoms to the skin of molecule B in places where a water molecule is likely to be found. These additional centers in the skin of B will be assigned a much lower weight then surface atoms of B. By doing so, overlaps of such atoms with the core of A will be less penalizing but overlaps with the skin of A will contribute somewhat to the score, effectively representing an "optional" atom.

3.2 Affinity Function Scoring

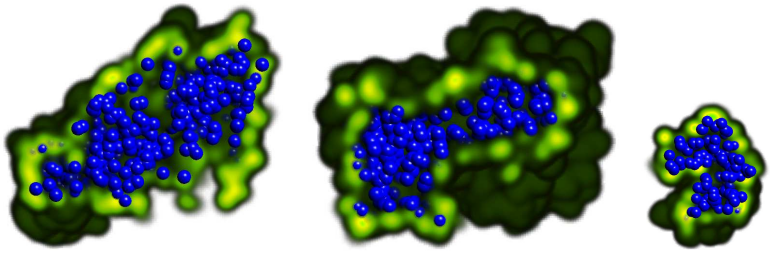
Our scoring function is based on the grid scoring approaches of [9,10,19]. However, the fundamental difference is that the new score is based on *functional* interactions between the various skins and cores. By using positive real values as the weights for the smooth particle representation of the affinity function defined over the skin and imaginary values in the representation of the core regions, we will yield negative numbers for core-core overlaps and positive numbers for skin-skin overlaps during the convolution. In addition, we can define a finite number of groups of centers in the smooth particle representations, each group having its own weight. The weight will directly affect the contribution of (the function placed on) these centers to the score, so the new weighted affinity function for the j^{th} molecule takes the form



(a) An immunoglobulin from the influenza virus (IGG1, LAMBDA) (PDB:2VIR.PDB)

(b) Idiotypic FAB 730.1.4 (IGG1) of virus neutralizing antibody(PDB:1IAI.PDB)

(c) Hyhel-5 FAB (PDB:1BQL.PDB)



(d) Hemagglutinin (PDB:2VIR.PDB)

(e) Anti-Idiotypic Fab 409.5.3 (Igg2A)(PDB:1IAI.PDB)

(f) Bob-white Quail Lysozyme (PDB:1BQL.PDB)

Figure 3: We show three examples of populating the outer skin region and detecting surface atoms for three antibody antigen complexes. In the first row, we show one of the molecules of the complex where we populate the outer region, and in the second row, the other molecule where we detect surface atoms to form the second skin. The first row contains the grown skin layer shown in red, with higher densities being yellow and green. The surface skin in the second row is shown in green. We show a cut away to reveal the two skins. In the first row, the three molecules/skins had 3263/4519, 3342/4555, 3243/4308 atoms/kernel centers respectively. In the second, there were 988/1087, 4956/1719 and 5293/469 surface and interior atoms respectively.

$$Q^j(\vec{x}) = \sum_{k=1}^{M} \gamma_k^j \phi_k(\vec{x}) = \sum_{k=1}^{M} \gamma_k^j \phi(\vec{x} - \vec{x}_k).$$

such that $\phi(\vec{x}) = e^{(\frac{\mathcal{B}\|\vec{x}\|_2^2}{R_k^2} - \mathcal{B})}$ and $R_k = 1$. With this new definition of molecular shape affinity we can weight the Gaussians to reflect Core-Core clashes or Skin-Skin overlaps. For any two Molecules A & B to be docked, the parameters γ_k^j takes the form

$$\gamma_k^A = \begin{cases} 1 \sim skinA \\ \rho i \sim coreA \end{cases}, \gamma_k^B = \begin{cases} 1 \sim skinB \\ \rho i \sim coreB \end{cases}$$

The translational convolution search scoring then becomes:

$$P(x, y, z) = (Q^A \otimes Q^B)(x, y, z) = \int_{\tau_1} \int_{\tau_2} \int_{\tau_3} Q^A(\tau_1, \tau_2, \tau_3) Q^B(x - \tau_1, y - \tau_2, z - \tau_3) \; d\tau_1 d\tau_2 d\tau_3 \qquad (2)$$

Making γ_k positive causes the scoring function to add SkinA-SkinB overlaps positively, while setting γ_k to be imaginary causes CoreA-CoreB interactions to clash negatively. However, we shall not construct the N^3 grid directly and perform the convolution using the NFFT. The real component of the score corresponds to the summation of the positive skin-skin and the negative core-core overlaps. The imaginary part corresponds to the cumulative overlaps between skin and core regions.

3.3 Three Dimensional Translation Search

Our approach requires significantly less memory than previous Fourier methods by obtaining a compressed representation of the convolution profile of the density maps while retaining high accuracy. This compressed representation can be computed directly with the NFFT algorithm, which allows us to compute the translation search from a reduced set of frequencies in $\mathcal{O}\ (M \log M)$, where M is the number of atoms, instead of the $\mathcal{O}(N^3 \log N)$ operations required for the direct FFT.

The first step of the NFFT based fast corrolation method is to replace the smooth particle map with an accurate periodic form such that a Fourier series can be used to approximate $Q^A(\vec{x})$. To this end, we translate, rescale and embed the particle data located at $\{\vec{x}_i\}_{i=1}^M$ to fit in a volumetric interval $[-0.25, \ldots, 0.25)^3 = [-0.25, \ldots, 0.25) \times [-0.25, \ldots, 0.25) \times [-0.25, \ldots, 0.25)$. This is achieved by computing the parameters $p_1 = max_{i,j}(\|\vec{x}_i - \vec{x}_j\|)$, $p_2 = \arg\max_{\vec{x}} \left\{ e^{\frac{\mathcal{B}\|\vec{x}\|_2^2}{R_k^2}} \leq 10^{-16} \right\}$, $p = p_1 + 2p_2$ and the center $\vec{c} = \frac{\sum_{i=1}^M \vec{x}_i}{M}$. Now, re-center and rescale the particle centers $\{\vec{x}_i\}_{i=1}^M$ such that

$$\vec{x}_i := \frac{\vec{x}_i - \vec{c}}{p}$$

for all $i = 1 \dots M$. The second step is to rescale and truncate the kernel $\mathcal{K}(x)$ s.t.

$$\mathcal{K}_T(\vec{x}) = \mathcal{K}\left(\frac{\vec{x}}{p}\right)\mathcal{X}\left(\frac{\vec{x}}{p}\right)$$

where

$$\mathcal{X}(\vec{x}) = \begin{cases} 1 & \text{if} \quad -p \leq \vec{x} < p \\ 0 & ow. \end{cases}$$

Under this translation and rescaling we replace Q with

$$Q^A(\vec{y}) := \sum_{j=1}^{M} \gamma_j \mathcal{K}_T(\vec{y} - \vec{x}_j).$$

This truncated and rescaled form allows us to represent $Q^A(\vec{x})$ with a Fourier series approximation in Π^3 for n desired frequencies for some error ϵ. The choice on the number of computed frequencies will control the error of the approximation.

The next step is to expand the kernel function into its Fourier series, as $\mathcal{K}_T(\vec{x}) = \sum_{k \in I_n} h_k e^{2\pi i \vec{x} \cdot k} + \epsilon$ for all $\vec{x} \in \Pi^3$ for some error ϵ. The index I_n refers a volumetric grid of truncated frequencies (i.e. $I_n = \{k = (k_1, k_2, k_3) \in \mathbb{Z}^3 : -n/2 \leq k_i < n/2\}$). Notice that for $n \to \infty$ the equality will be exact in an l_2 sense. Moreover, the equality will be pointwise for all kernels that satisfy the Direchlet conditions, except at the discontinuity points. With this Fourier series representation

$$\begin{aligned} Q^A(\vec{x}) &= \sum_{j=1}^{M} \gamma_j \mathcal{K}_T(\vec{x} - \vec{x}_j) = \sum_{j=1}^{M} \gamma_j \left(\sum_{k \in I_n} h_k e^{2\pi i (\vec{x} - \vec{x}_j) \cdot k} \right) \\ &= \sum_{k \in I_n} h_k e^{2\pi i x \cdot k} \sum_{j=1}^{M} \gamma_j e^{-2\pi i \vec{x}_j \cdot k} = \sum_{k \in I_n} \alpha_k h_k e^{2\pi i x \cdot k}, \end{aligned}$$

where $\alpha_k = \sum_{j=1}^{M} \gamma_j e^{-2\pi i (\vec{x}_j) \cdot k}$. An approximation of $Q(\vec{x})$ can be obtain by just computing a finite number of frequenices n i.e.

$$Q^A(\vec{x}) = \sum_{k \in I_n} \alpha_k h_k e^{2\pi i x \cdot k} + \varepsilon_1$$

for some error ϵ_1. Following the same procedure for $Q^B(\vec{x})$ we obtain

$$Q^B(\vec{x}) = \sum_{k \in I_n} \beta_k h_k e^{2\pi i x \cdot k} + \varepsilon_2.$$

The approximation $\hat{P}$ to the convolution integral (2) now becomes

$$\begin{aligned}\hat{P}(x,y,z) &= \int\limits_{(\tau_1,\tau_2,\tau_3)\in\Pi^3} Q^A(\tau_1,\tau_2,\tau_3)Q^B(x-\tau_1,y-\tau_2,z-\tau_3)\ d\tau_1 d\tau_2 d\tau_3 \\ &= \int\limits_{(\tau_1,\tau_2,\tau_3)\in\Pi^3} \left(\sum_{k\in I_N} \alpha_k h_k e^{-2\pi i\vec{\tau}\cdot k}\right) \sum_{k'\in I_N} \beta_{k'} h_{k'} e^{2\pi i(\vec{x}-\vec{\tau})\cdot k'} \\ &= \sum_{k\in I_N} \alpha_k\beta_k h_k^2 e^{2\pi i\vec{\tau}\cdot k}.\end{aligned} \tag{3}$$

The last step is true since $\int\limits_{-\frac{1}{2}}^{\frac{1}{2}} e^{2\pi i\tau_j(k_j-k_j')}d\tau_j = 1, \quad j = 1\ldots 3$ if $k_j = k_j'$ and zero otherwise.

Fast Translation Convolution Algorithm

1. ***Preprocessing:***
   (a) Compute

$$p_1 = max_{i,j}(\|\vec{x}_i - \vec{x}_j\|), \quad p_2 = \arg\max_{\vec{x}} \left\{ e^{\frac{\mathcal{B}\|\vec{x}\|_2^2}{R_k^2}} \leq 10^{-16} \right\}, \quad p = p_1 + 2p_2$$

   (b) Rescale and truncate the kernel $\mathcal{K}(x)$ s.t.

$$\mathcal{K}_T(\vec{x}) = \mathcal{K}\left(\frac{\vec{x}}{p}\right)\mathcal{X}\left(\frac{\vec{x}}{p}\right)$$

   where

$$\mathcal{X}(\vec{x}) = \begin{cases} 1 & \text{if} \quad -p \leq x_i < p, \quad i = 1\ldots 3 \\ 0 & ow. \end{cases}$$

   (c) Compute

$$c = \frac{\sum_{i=1}^M x_i}{M}$$

   (d) Re-center and rescale particle centers $\{\vec{x}_i\}_{i=1}^M$ and output points $\{\vec{y}_i\}_{i=1}^N$ such that

$$\vec{x}_i := \frac{\vec{x}_i - c}{2p}$$

   for all $i = 1\ldots M$ and $j = 1\ldots N$.

(e) For each $k \in I_n$ compute the Fourier series integrals

$$h_k = \int_{\Pi^3} \mathcal{K}_T(\vec{x}) e^{-i2\pi\vec{x}\cdot k} \; d\vec{x}.$$

2. ***Inputs:*** The variables $\{N, n\}$, where N relates to the accuracy of the Fourier series representation of the smooth particle data and n is the index for the number of frequencies that shall be computed in the N point DFT. Other variables include the smooth particle data:

$$Q^A(\vec{x}) = \sum_{j=1}^{M} \gamma_j^A e^{(\frac{\mathcal{B}\|\vec{x}-\vec{x}_j\|_2^2}{R_j^2} - \mathcal{B})} \quad \text{and} \quad Q^B(\vec{x}) = \sum_{j=1}^{M} \gamma_j^B e^{(\frac{\mathcal{B}\|\vec{x}-\vec{x}_j\|_2^2}{R_j^2} - \mathcal{B})}.$$

where we assume all the atomic radii to be the same.

3. ***Preprocessing:*** For each $k \in I_n$ compute the frequencies of the kernel function

$$h_k = \frac{1}{N^3} \sum_{k' \in I_n} e^{-k'/N} e^{-2\pi i k' \cdot k/N}.$$

For tensor products kernels ϕ (i.e. Gaussian), we can compute the above equation with a single 1D FFT of length N with a computational cost of $\mathcal{O}(N \log N)$ and $\mathcal{O}(N)$ storage. An alternative for non tensor product radial symmetric kernels is to compute the Fourier transforms analytically. This process is greatly simplified since the Fourier transform of an radially symmetrical function is itself radially symmetrical. However, we have to take into account the aliasing error.

4. ***Truncated DFT:***
With the NFFT$'$ compute the first $k \in I_n$ frequencies of N point DFT of the particle data

$$\alpha_k^{'} = \sum_{j=1}^{M} \gamma_j^A e^{-2\pi i(\vec{x}_j)\cdot k} \quad \text{and} \quad \beta_k^{'} = \sum_{j=1}^{M} \gamma_j^B e^{-2\pi i(\vec{x}_j)\cdot k}.$$

This step provides the frequencies directly from the atomic centers, which is a very sparse map. By avoiding using the FFT on a large grid (slow and high storage cost) or evaluating the DFT polynomial ($\mathcal{O}(Mn^3)$ computational cost). The DFT can approximated with a m^{th} order NFFT$'$ with a computational cost is $\mathcal{O}(\alpha^3 M \log M + (2m+1)^3 M)$ and $\mathcal{O}(\alpha^3 M)$ storage.

5. ***Convolve:*** For each $k \in I_n$ compute

$$c_k = \alpha_k^{'} \beta_k^{'} h_k^2.$$

By multiplying the frequencies α_k and β_k, the approximate convolution of the maps Q^A and Q^B is performed. The point wise multiplication cost is $\mathcal{O}(n^3)$ and storage is $\mathcal{O}(n^3)$.

6. ***Inverse DFT:*** With the NFFT Compute

$$\hat{P}(\vec{x}_j) = \sum_{k \in I_n} c_k e^{2\pi i \vec{x} \cdot k}.$$

The NFFT algorithm produces an output in the form of m^{th} order cardinal B-splines. Thus is we sample the map $\hat{P}$ $\bar{N}$ times, the total computational cost is $\mathcal{O}(n^3 \log n + (2m+1)^3 \bar{N})$ and storage is $\mathcal{O}(n^3)$. The total computational cost of this algorithm, excluding the preprocessing step, is $\mathcal{O}(\alpha^3 M \log M + n^3 \log n + (2m+1)^3 \bar{N})$ and $\mathcal{O}(\max(\alpha^3 M, \beta^3 M) + N)$ storage.

Note that for many docking problems, we can set R_k to the same constant. However, for many proteins R_k can be separated into a group of about 15 distinct values. This means, the step in the NFFT$'$ algorithm can be broken down into $M/15$ groups with the same frequency size n. This leads to 15 FFT extra steps. However, the FFT step is computational faster than the blurring step, thus there is no discernible timing difference.

Inverse step In the inverse step, we need to find the position of the peak given the product of frequencies. One solution to this problem is to append zeros to the frequencies map and take the inverse fourier transform. This has both a high space and time complexity which we would like to avoid. Another solution is to first perform an inverse fourier transform and do a sync interpolation around local peaks. This is again computationally expensive.

We follow a heuristic where we explore only those regions which look promising in a two step process. In the first step, we invert the product map to get a low resolution profile. From this profile, we store the first few peak positions. In our experiments, we used 10 peaks. In the second step, using the NFFT algorithm, we convert the product map to a set of B-Splines of order m_2. Using the location information from the first step, we locally expand the B-Spline grid around only those regions and search for the top peaks. We use the FFT to compute the local expansions as they are a convolution of B-Spline functions.

3.4 Three Dimensional Rotational Search

For each rotational step the 3D Translation algorithm is performed and a predetermined number of maximum correlations are conserved. Let $R_s = \{R \in SO(3)\}$ be a set of N_R rotations.

Full Search Algorithm

1. ***Inputs:*** The variables N, n and the set of all predefined rotations R_s,

$$Q^A(\vec{x}) = \sum_{j=1}^{M} \gamma_j^A e^{(\frac{\mathcal{B}\|\vec{x}-\vec{x}_j\|_2^2}{R_j^2} - \mathcal{B})} \quad \text{and} \quad Q^B(\vec{x}) = \sum_{j=1}^{M} \gamma_j^B e^{(\frac{\mathcal{B}\|\vec{x}-\vec{x}_j\|_2^2}{R_j^2} - \mathcal{B})}.$$

The first molecule Q^A is fixed and the second molecule Q^B will be rotated. For every rotation of Q^B, the 3D translational scoring of Q^A and Q^B will be computed and the maximum obtained.

2. ***Preprocessing:*** For each $R \in R_s$ compute

   (a) ***Rotation:***

$$Q^A(\vec{x}) = \Lambda_R \left(\sum_{j=1}^{M} \gamma_j^A e^{(\frac{\mathcal{B}\|\vec{x}-\vec{x}_j\|_2^2}{R_j^2} - \mathcal{B})} \right) = \sum_{j=1}^{M} \gamma_j^A e^{(\frac{\mathcal{B}\|\vec{x}-\Lambda_R(\vec{x}_j)\|_2^2}{R_j^2} - \mathcal{B})}$$

   (b) ***NFFT based Convolution***

$$\hat{P}_R(x, y, z) = Q^A \hat{\otimes} Q^B,$$

   where $\hat{\otimes}$ indicates the approximate NFFT based fast translational convolution algorithm.

3. ***Maximum:*** For all rotations in R_s and translation location $\vec{x}$ compute

$$\arg\max_{\vec{x},R} \hat{P}_R(x, y, z).$$

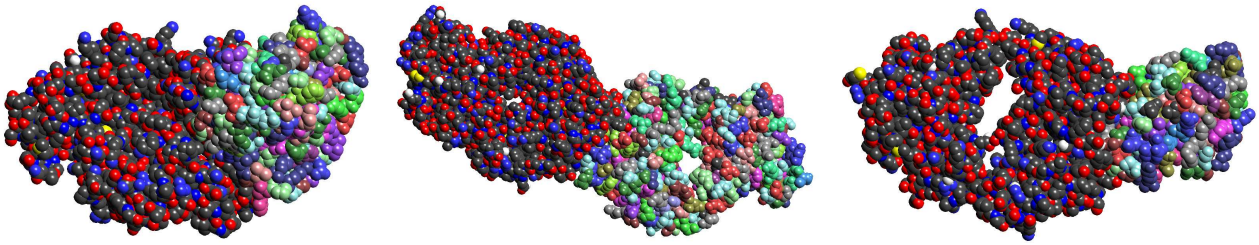
   For a total of r_s rotations the total computational cost is $\mathcal{O}(r_s(\alpha^3 M \log M + n^3 \log n + (2m+1)^3 \bar{N} + N \log N))$ and $\mathcal{O}(\max(\alpha^3 M, \beta^3 M) + N)$ storage.

For an exhaustive rotational search the number of orientations will be of the order of $r_s = N^3$, where N represents the number of steps in a 1-D full rotation. Due to the smoothness of the convolution operator we sample the output profile at $\bar{N} = M$ points. Since we need $\mathcal{O}(M \log M)$ computations for each of the rotational steps, the total complexity cost for a full search is $\mathcal{O}(N^3 M \log M)$ and $\mathcal{O}(M)$ storage requirements. Notice that for a grid based methods like Zdock [10], each translational search requires $\mathcal{O}(N^3 \log N)$ for a N^3 grid. Thus a total of $\mathcal{O}(N^6 \log N)$ operations and $\mathcal{O}(N^3)$ storage are needed. –A comparison with the most popular FFT and/or Spherical Harmonic methods is shown in the Appendix A.–

4 Results of our Protein-Protein Docking

In this section we report on the computational efficiency of our fast adaptive docking method on three complexes: Hyhel-5 fab complexed with bobwhite quail lysozyme (PDB:1BQL.PDB), Idiotype-anti-idiotype fab complex (PDB: 1IAI. PDB) and an influenza virus hemagglutinin complexed with a neutralizing antibody (PDB: 2VIR .PDB). We will simply refer to these complexes as **complex 1, complex 2, complex 3** respectively (see figure 4). In all experiments, we perform the translational search using the NFFT based method for a fixed set of orientations.

Shape complementarity is the only term in our scoring function, for this experiment. This is a good comparison given that the fast Fourier method is the best known algorithm for convolution in terms of speed and memory. Our results demonstrate faster docking with lower memory requirements while providing accurate results.



(a) **Complex 1:**Hyhel-5 fab complexed with bobwhite quail lysozyme (PDB:1BQL.PDB)

(b) **Complex 2:**Idiotype-anti-idiotype fab complex(PDB:1IAI.PDB)

(c) **Complex 3:**influenza virus hemagglutinin complexed with a neutralizing antibody (PDB:2VIR.PDB)

Figure 4: The three complexes we have used as test cases. The first molecule is colored using standard atom colors while the atoms in the second molecule are colored by their residue type to differentiate the two molecules in the complex.

Test Conditions For all three complexes we shall compare our method to the FFT grid based approach to test the accuracy of our NFFT fast search algorithm. The molecules are embedded in a grid of 128^3. Zero padding is also done to avoid any warp around during the convolution step.

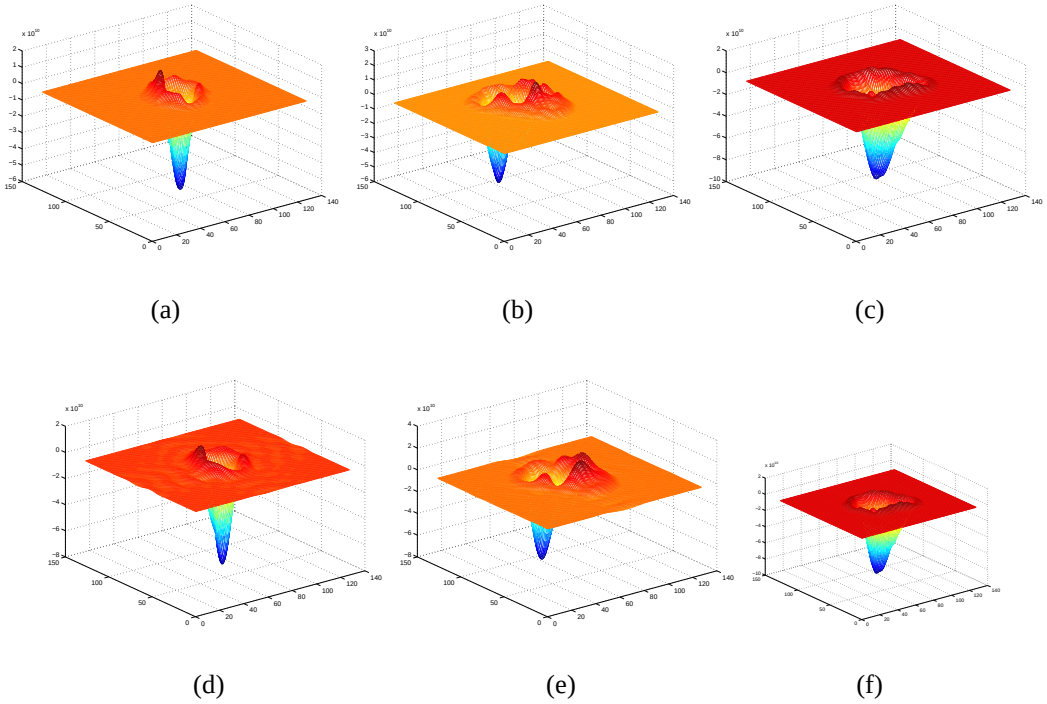
We define the **Profile** as a $P[\cdot,\cdot,\cdot]$ matrix, where each element represents the overlap of the two molecules for a unique translation. **Energy E_P of the profile** is the norm: $E_P = \|P[\cdot,\cdot,\cdot]\|_{l_2}$.

All experiments where performed on a Sun E25K with 128 processors and 512 G of shared memory. Only one processor was used for a translational search. Different orientations are computed in parallel. The FFT was performed with the optimized flag of the Fastest Fourier Transform of the West (FFTW) package [18],

Energy retained in profile The energy retained in the profile is a value the users can specify to determine the number of fourier series coefficients they need to use. We tabulate the energy retained for different rates of decays of the input Gaussians for the three test cases. It is seen that as we go to a lower resolution, fewer number of frequencies are required to obtain the same accuracy, providing one method of performing a hierarchical docking search.

Number of freq.	$\beta = -0.5$		$\beta = -1$	
	l^2	l^∞	l^2	l^∞
16^3	6.3364	3.0409	9.9454	3.5909
20^3	3.9761	1.2994	7.9016	1.7434
32^3	1.1991	0.2889	5.3285	0.5909

Table 1: Fraction of energy lost, in %, for **complex 1**, with $\alpha = m = 2$ as the NFFT parameters



(a) (b) (c)

(d) (e) (f)

Figure 5: A slice of the profiles containing the peak for a given orientation for each of the three test cases is shown. In the top row, the FFT profiles are given while the corresponding Fast docking results are given below.

The convolution of smooth functions like the electron density yields a profile that is largely composed of low frequencies. In Figure 5 we show one slice of each of the three profiles of our test cases where the maximum was found. Notice that since most of the energy of the profile is located at the lower frequencies (Table 1, 2 and 3), most of the profile can be reconstructed with a few frequencies.

Full Rotational Search The rotational search is currently a full 3 degree search. We present results for a low sampling of the space. For each of the three test cases, we used 128^3 FFT results to compare to. We performed the search using only structure as the affinity function. In each case, we used 16^3 frequencies and $\alpha = m = 2$ as the NFFT parameters. In table 4 we present the average deviation of the closest in the top ten of our peaks to the peak of the FFT method.

Since we are currently using only the shape or electron density as our scoring function, the docking position is limited to shape complementarity. Other important scoring functions like hydrophobicity matching and electrostatics complementarity will be considered in future experiments.

Number of freq.	$\beta = -0.5$		$\beta = -1$	
	l^2	l^∞	l^2	l^∞
16^3	4.5203	3.5743	6.8897	4.2208
20^3	2.5131	1.4592	5.1096	1.8793
32^3	0.8462	0.2480	3.6941	0.5297

Table 2: Fraction of energy lost, in %, for **complex 2**, with $\alpha = m = 2$ as the NFFT parameters

Number of freq.	$\beta = -0.5$		$\beta = -1$	
	l^2	l^∞	l^2	l^∞
16^3	4.8228	2.0457	7.7806	2.3983
20^3	2.7570	0.8029	6.0601	1.0721
32^3	0.9504	0.2017	4.6343	0.4111

Table 3: Fraction of energy lost, in %, for **complex 3**, with $\alpha = m = 2$ as the NFFT parameters

Test case	Average error in peaks (Å)
Complex 1	2.64
Complex 2	3.10
Complex 3	3.11

Table 4: The average deviation of the peak in the full FFT method from the closest in the top ten of our method, for each orientation, is tabulated for each of three cases. Since we have performed only a coarse rotational sampling, there are orientations where the peak of the FFT is present in a slightly different orientation for our method. We also do not weight the average by the value of the peak in the FFT method. This results in some orientations where the distance between the peaks are large, leading to the large average deviation. The rate of decay of the kernel functions β was set as -1.0. The NFFT parameters used are $\alpha = m = 2$ and 16^3 fourier series coeffi cients were computed.

Timings The FFT works on a grid. Hence we need to discretize our input data to a grid. The convolution requires the grid to be a power of 2 in each dimension, and must be large enough to accommodate both molecules. Thus, zero padding which typically will double the grid size is needed. For average size proteins (such as the superoxide dismutase) and grid spacing of 1A for reasonable resolution, a volume of 256^3 is necessary. For docking of two larger molecules ($>$70 Åradius), one would need to go to grids beyond 512^3. This is clearly a very expensive operation with respect to time consumed. When we deal with flexible molecules, the need to perform Fourier transforms through the pipeline prohibits the use of the FFT.

For our search NFFT based method, the low resolution frequencies can be obtained efficiently as shown in Table 5. The second step of the NFFT based method is to run the NFFT and obtain the coefficients of the Cardinal B-splines that describes the convolution profile. This step is significantly faster than the NFFT$'$ step. The third is to actually to compute the maximum from the profile B-splines. If a low resolution grid of size M is used, by using an FFT of size M we compute the inverse significantly faster than Step 1 with a 1 % Translation location error.

Frequencies, (α, β)	$\alpha = 2, \beta = 2$	$\alpha = 2, \beta = 3$	$\alpha = 2, \beta = 4$	$\alpha = 4, \beta = 4$	FFT(256^3)
1000	0.077414	0.142723	0.254004	0.347024	16.798823
4096	0.114369	0.182574	0.298170	0.662981	16.798823
8000	0.170260	0.240088	0.360787	1.214280	16.798823

Table 5: Time in seconds taken to estimate Fourier coeffi cients with the NFFT′ for different oversampling factors α and β for a molecule with 1100 atoms. The time to perform the FFT for a 256^3 grid is also given. Note that the FFT was performed with the FFTW with the optimized flag on.

This is clearly a very expensive operation with respect to time consumed. When we deal with flexible molecules, the need to perform Fourier transforms through the pipeline prohibits the use of the fast Fourier transform. We see that even a FFT of a very low-resolution model of 128^3 is more time consuming than our method with sampling factors $\alpha, \beta = 2, 2$.

Memory requirements The experimental results closely followed the theoretical memory requirement that is linear in the number of expansion points. We used a memory over-sampling factor of 2. Hence for our three test cases which had approximately 10000 to 15000 expansion points, we needed approximately 5MB of space. This is in contrast to 268 MB for a 256^3 grid for the FFT Grid Based approaches. This is also very low compared with the memory requirements of other methods discussed in Appendix A.

5 Conclusion

In this paper we introduce an adaptive irregular spaced Fourier method based on grid free smooth particle representations to efficiently predict protein-protein docking sites. Our algorithm is significantly faster than the grid based FFT docking algorithms by avoiding the construction of the volumetric grid. In the future, we envision improving the speed, efficiency, generality and flexibility of predicting, visualizing, and analyzing protein-protein interactions with significantly more degrees of freedom.

In the current form of the algorithm we have only docked shape with the electron density. To further improve the accuracy of our predictions we shall incorporate and calibrate our algorithm with associated molecular properties such as electrostatics and hydrophobicity.

In section 4 the rotational search is discretized in uniform Euler angles, a better choice is to use the optimized uniform sampling described in [27].

The choice for N, n, m_1, m_2, α_1 and α_2 are not provided in this paper. Correctly choosing this parameters a-priori can provide an optimal balance between accuracy of the compressed profile and speed of computation. To achieve the balance, precise error bounds for our method are needed. In particular, for a given map resolution we shall derive estimates for N, n, m_1, m_2, α_1 and α_2 such that the error of the convolution profile is bounded by user given threshold.

To increase the precision of the maximum search in our compressed convolution profiles, we shall develop a local interpolation scheme to compute the convolution profile in an accurate and fast manner. Moreover, more efficient non convex optimization schemes for peak detection shall be investigated.

Finally, we shall refine and calibrate our docking procedures and validate docking computations on a set of known complexes, and subsequently on challenge problems from CAPRI, and from collaborations with experimental scientists.

Acknowledgments

We wish to thank Dr. Art Olson and Dr. Michel Sanner of The Scripps Research Institute for their invaluable feedback during the development of this algorithm.

Appendix A: Computational Analysis of Previous Approaches

The method we develop in this paper is based on the grid FFT approaches. However, since are method is grid free we significantly reduce the memory and time complexities. We are interested how this method also measures with respect to more modern Docking approaches .

Current docking methods can be characterized into Fourier and/or Spherical harmonic approaches. To make a fair comparison with these newer methods a complete complexity analysis is presented. In particular, we are interested in complexity analysis of the FFT/Spherical harmonic approach of Kovacs *et al* and the Spherical harmonic method of Ritchie *et al*.

A.1 FFT/Spherical Harmonic approach

The algorithm developed by Kovacs *et al* [24] relies on relating the spherical harmonic expansion coefficients on a sphere of radius r to a Discrete Fourier Transform (DFT) representation by taking the Fourier transform of the correlation function of (Equation (3) in [24]. Let $f : \mathbb{R}^3 \to \mathbb{R}$ and $g : \mathbb{R}^3 \to \mathbb{R}$ be two density functions which are bounded and of compact support inside a volume $V \subset \mathbb{R}^3$. Expanding f by series of spherical harmonic expansions on a sphere of radius r we obtain:

$$f(ru) \approx \sum_{l=0}^{B-1} \sum_{-l}^{l} \hat{f}_{lm}(r) Y_{lm}(u) \tag{A-1}$$

where Y_{lm} are spherical harmonics, B the order of the expansion and u is a unit vector. Moreover, let $\Lambda_R f$ be a rotational operator in Euler angle representation (ϕ, θ, ψ). For a rotation R on f we have:

$$\Lambda_R f(ru) = \sum_{l=0}^{B-1} \sum_{m=-l}^{l} \sum_{n=-l}^{l} \hat{f}_{lm} D^l_{mn}(\phi, \theta, \psi) Y_{lm}(u) \tag{A-2}$$

where

$$D^l_{mn}(\phi, \theta, \psi) = e^{-im\phi} d^l_{mn}(\theta) e^{-in\psi},$$

and

$$d^l_{mn}(\theta) = (-1)^m \sqrt{\frac{(l-m)!}{(l+m)!}} P^m_l(cos\theta).$$

Finally, let T_ρ be the translation operator such that

$$T_\rho g(x, y, z) := g(x, y, z - \rho) \quad \forall (x, y, z) \in \mathbb{R}^3$$

The correlation function can now be built from the density functions (f, g) with their respective rotations (R, R') and intermolecular distance translation operator T_ρ. This leads us to the 6D degrees of freedom correlation

$$c(R, R'; \rho) = \sum_{ll'mm'nn'} \int_{R^3} \overline{\Lambda_R f} \cdot \overline{T_\rho \Lambda_{R'} g}. \tag{A-3}$$

Replacing the operators $(\Lambda_R, \lambda_{R'}, T_\rho)$ into Equation (A-3) and making the following change of variables

$$\sigma = \phi - \frac{\pi}{2}, \quad \eta = \pi - \theta, \quad \omega = \psi - \frac{\pi}{2}$$

$$\sigma' = \phi' - \frac{\pi}{2}, \quad \eta' = \pi' - \theta, \quad \omega' = \psi' - \frac{\pi}{2}$$

and letting $\sigma = \eta - \eta'$ we obtain the following correlation function

$$
\begin{aligned}
c(\phi,\theta,\psi,\phi',\theta',\psi',\rho) &= \sum_{l=0}^{B-1}\sum_{l'=0}^{B-1}\sum_{m=-l}^{l}\sum_{n=-l}^{l}\sum_{m'=-l'}^{l'}\sum_{h'=-l'}^{l'}\sum_{h=-l}^{l}(-1)^n \\
&\cdot\ d^l_{nh}d^l_{hm}d^{l'}_{-nh'}d^{l'}_{h'm'}e^{i(n\sigma+h\eta+m\omega+h'\eta'+m'\omega')}I^{ll'}_{mnm'}(\rho). \\
&=: T(\sigma,\eta,\omega,\sigma',\eta',\omega',\rho)
\end{aligned} \tag{A-4}
$$

where

$$
I^{ll'}_{mnm'}(\rho) = \sqrt{(l+\frac{1}{2})(l'+\frac{l}{2})}\cdot\int_0^\pi\left[\int_0^\infty \overline{\hat{f}_{lm}(r)\hat{g}_{l'm'}(r')r^2\ dr}\right]\cdot d^l_{n0}(\beta)sin\beta\ d\beta.
$$

The previous correlation function is now in terms of the five Euler angles $\sigma,\eta,\omega,\eta',\omega'$ and the intermolecular distance ρ. Taking the Fourier transform of the previous equation leads to

$$
\hat{T}(n,j,m,h',m',\rho) = (-1)^n\sum_{l=0}^{B-1}\sum_{l'=0}^{B-1} d^l_{nh}d^l_{hm}d^{l'}_{-nh'}d^{l'}_{h'm'}I^{ll'}_{mnm'}(\rho). \tag{A-5}
$$

The correlation function in the sample domain is computed by taking an Inverse Fast Fourier Transform (IFFT) of equation A-5. This leads to a $2B\times 2B\times 2B\times 2B\times 2B$ Cartesian grid for the five angles $(\sigma,\eta,\omega,\eta',\omega')$ and one fixed singular intermolecular distance ρ. It is easy to see from equation (A-5) that $\mathcal{O}(B^5)$ Fourier coefficients are needed. Moreover, each entry requires B^2 computations thus the total computational cost for the IFFT is $\mathcal{O}(B^7\log B)$. Suppose that ρ is discretized into D_ρ steps and r is discretized D_r steps, then the total memory cost is $\mathcal{O}(B^5)$ and the total computational cost is $\mathcal{O}(D_rD_\rho B^7\log B)$

We can immediately observe several drawbacks to this algorithm. The most important being a lack of an error bound on the spherical harmonic representation. Moreover, the width of the discretization of the Euler angles is directly related to the number of spherical harmonic expansions. This is significant, since the output of the correlation function is much smoother than the resolution of the data. This implies that the discretization of the Euler angles can be made larger than the resolution of the original molecules with significant loss of accuracy.

The algorithm is too rigid for adaptively solving the correlation function. For example, if we require a high precision discretization of the Euler angles around a small region, then we are forced to solve it everywhere. With $\mathcal{O}(B^5)$ entries the memory requirements quickly become prohibitive. In practice about $128B^5$ bytes are needed. For $B=32$, the memory requirements become larger that 4 GB, placing well above many of today workstations. This motives discussing the following flexible approach for fast scoring by Ritchie on the double skin layer model.

A.2 Double skin layer Approach

This method relies on correlation scoring of the overlap between the double skin later model of both the ligand and the receptor molecules. This model involves to volumetric skins of the molecule. The interior skin is the union of all the van Der Waals volumes of the surface atoms. The density of this skin is represented as

$$\tau(\underline{r}) = \begin{cases} 1; & \underline{r} \in \text{ Surface atom} \\ 0; & \text{otherwise} \end{cases} .$$

The exterior skin is defined using the solvent-accessible and molecular surfaces and the density is represented as

$$\tilde{\rho}(\underline{r}) = \begin{cases} 1; & \underline{r} \in \text{ Surface skin} \\ 0; & \text{otherwise} \end{cases} .$$

The correlation functions are expanded in terms of spherical harmonics. However, in contrast to Kovac's method, the skin is represented with real spherical harmonics and radial functions. Expanding the inner skins we obtain

$$\tau(\underline{r}) \approx \sum_{n=1}^{N} \sum_{l=0}^{n-1} \sum_{m=-l}^{l} a_{nlm} R_{nl}(\underline{r}) y_{lm}(\theta, \phi); \quad n > l \geq |m| \geq 0$$

where $R_{nl}(\underline{r})$ is based on generalized Laguerre polynomials and N is the order of expansion. Please refer to [33] for the different choices for $R_{nl}(\underline{r})$. The total number of coefficients in this expansion is $\mathcal{O}(N^3)$. The spherical harmonic representation can be easily rotated around the Euler angles (α, β, γ). The updated rotated coefficients a'_{nlm} are computed as

$$a'_{nlm} = \sum_{m'=-l}^{l} a_{nlm'} D^l_{mm'}(\alpha, \beta, \gamma), \tag{A-6}$$

where

$$D^l_{mm'}(\alpha, \beta, \gamma) = e^{-im'\alpha} d^l_{m'm}(\beta) e^{-im\gamma}.$$

Also, let a''_{nlm} be the translated coefficients by the operator T_ρ [33].

In the double skin model the scoring function $S(\cdot)$ takes the form

$$S = \int \tilde{\rho}_A(\underline{r}_A) \tau_B(\underline{r}_B) \ dV + \int \tau_A(\underline{r}_A) \tilde{\rho}_B(\underline{r}_B) \ dV - Q \int \tau_A(\underline{r}_A) \tau_B(\underline{r}_B) \ dV$$

where the final term acts as a steric penalty for the interior-interior skin. The scoring function $S(\cdot)$ is written in terms of the Euler angles $\beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2$ and the intermolecular distance ρ:

$$S(\rho, \beta_1, \gamma_1, \alpha_2, \beta_2, \gamma_2) = \sum_{m=-L}^{L} Q^+_m \cos m\alpha_2 + Q^-_m \sin \bar{m}\alpha_2; \qquad L = N - 1, \tag{A-7}$$

where

$$Q_m^+(\rho,\beta_1,\gamma_1,\beta_2,\gamma_2) = \sum_{nlm'}^{N}(A_{nlm'}^{'\rho}(\rho)b_{nlm'}^{'\tau}+A_{nlm'}^{'\tau}(\rho)b_{nlm'}^{'Q})\delta_{mm'}, \quad \text{(A-8)}$$

$$Q_m^-(\rho,\beta_1,\gamma_1,\beta_2,\gamma_2) = \sum_{nlm'}^{N}(A_{nlm'}^{'\rho}(\rho)b_{nl\bar{m}'}^{'\tau}+A_{nlm'}^{'\tau}(\rho)b_{nl\bar{m}'}^{'Q})\delta_{mm'}, \quad \text{(A-9)}$$

and

$$A_{n'l'm'}^{'\tilde{\rho}}(\rho)=\sum_{nlm}^{N}a_{nlm}^{'\rho}K_{nn'll'|m|}(\rho)\delta_{mm'}. \quad \text{(A-10)}$$

See [33] for the term $K_{nn'll'|m|}(\rho)$.

To make a complexity analysis of the algorithm, we assume first that all the six degrees of freedom discretized such that
Let

$$\begin{aligned} N^2 &= \text{Number of discrete steps of } (\beta_1,\gamma_1), \quad \text{(A-11)}\\ N^2 &= \text{Number of discrete steps of } (\beta_2,\gamma_2),\\ D_\rho &= \text{Discretization steps of } \rho,\\ D_r &= \text{Number of Spherical Harmonic shells}\\ M_{\alpha_2} &= N = \text{Discretization steps of } \alpha_2. \end{aligned}$$

The computational cost of Ritchie's algorithm can be easily determined from the following equations (Equations (7.17-7.18) in [33]). The cost to compute equation (A-10) involves $\mathcal{O}(N^3)$ computations. This implies that every time that (ρ,β_1,α_1) is updated, from equations (A-7), (A-8) and (A-9) $\mathcal{O}(N^5)$ computations are required. However, any update of (β_2,γ_2) only requires $\mathcal{O}(N^3)$. Moreover, due to the Fourier series representation of (A-8) we see that $\mathcal{O}(N)$ computations are required for any update of α_2.

This implies that the total computational cost is $\mathcal{O}(D_\rho N^7+D_\rho(N^2-1)N^5+D_\rho N^5(M_\alpha-1))$. However, the memory cost boils down to $\mathcal{O}(D_\rho N^3)$. The D_ρ factor is due the caching of the integral $K_{nn'll'|m|}(\rho)$. However, this step can be made apriori, therefore the computational complexity is reduced to $\mathcal{O}(D_\rho N^7+D_\rho(N-1)N^5+D_\rho N^5(M_\alpha-1))$. However, notice, that for even one single orientation calculation, $\mathcal{O}(N^5)$ computations are needed.

We can make a direct comparison between both methods and our own docking search algorithm. We shall first establish a common notation to directly compare both methods. Let $N=B$ (both methods having the same order). And let discretized all five rotational angles with the same step size. In Table 6, the total computational and memory costs for both methods are shown. Notice that we also place the complexity cost for the grid based method Zdock, see [10].

	Time complexity	Space Complexity
Ritchie *et al*	$\mathcal{O}(D_\rho N^7 + D_\rho(N-1)N^5 + D_\rho N^5(2N-1))$	$\mathcal{O}(D_\rho N^3)$
Kovacs *el al*	$\mathcal{O}(D_r D_\rho N^7 \log N)$	$\mathcal{O}(D_r N^5)$
Grid Based Docking (Zdock)	$\mathcal{O}(N^6 \log N)$	$\mathcal{O}(N^3)$
Our Method	$O(N^3 M \log M)$	$\mathcal{O}(M)$

Table 6: Complexity analysis results

From Table 6 we observe that Ritchie's method is better than Kovacs' with respect to asymptotic worst-case space and time complexity. In particular, Kovacs method is very memory intensive. However, Ritchie's method is based on spherical harmonics, thus ill suited for representing non star shaped molecules. In addition, no error estimators have been developed, thus the confidence on the accuracy of the results in lost. Our proposed method has significantly better asymptotic worst-case space and time complexity. It performs a combination of 3-D translational and 3-D rotational search. For each rotation a translational search is performed in O($M \log M$), where M of the order of the number of atoms in the largest molecule. Since N^3 rotations are performed then we can compute the full six dimensional search in O($N^3 M \log M$) time and O(M) memory. Finally, M is significantly smaller than N^3.

B Appendix B: Molecular Skin Population

We define the skin region of one molecule as the region belonging to the Solvent Accessible Surface Volume (V_{SAS}). Since we use the convolution of Gaussian functions over atom centers as our data structure for representing molecular structure, we define the skin implicitly as a set of spheres packing the region. The packing density is itself chosen to approximately equal the packing of the atoms belonging to the molecular surface.

The region is defined over a trilinear grid in which the molecule is embedded. The grid spacing h is chosen to preserve the features of the molecule. Assuming that the interatomic distance is $\sim 1\mathring{A}$, we can use $h = 0.5\mathring{A}$. By finding the boundary vertices of the SAS, we can obtain potential centers for the skin spheres. To prevent aliasing artifacts, we randomly choose potential centers to test for whether it should contain a sphere or not. A packing algorithm then decides, based on the packing density required, if a potential center should contain an atom or not.

Algorithm 1 Skin population

Procedure AddAtom(cell c)

$c' = \{cell \in G : dist(c', c) \leq 2r_p\}$
$A_s(c')$ = set of skin atoms in c'
if $A_s(c') \equiv \{\}$ **then**
  return true;
end if
return false;

Inputs are: $[M, A_i, r_i, \vec{c}_i, i = 1..M, r_p, h]$
Output is: Adaptive grid G with grid points classifi ed as SES or not.

{Construct adaptive octree for atoms}
for $i = 1$ to M **do**

  {Insert A_i into grid.}
  for all $g \in G : |c_i - g| \leq r_i + r_p$ **do**
    $g \leftarrow V_{SAS}$
  end for

end for

{Classify the boundary cells}
for all Cell $c \in G$ **do**
  $v_1..v_8 \leftarrow$ vertices of c.

  if $(\exists v_i \in V_{SAS}) \wedge (\exists v_j \notin V_{SAS}), i, j \in 1..8$ **then**
    $c \leftarrow S_{SAS}$
  end if

end for

{Add skin region spheres}
for all Cells $c \in S_{SAS}$, chosen randomly **do**

  if AddAtom(c) **then**
    New Skin Probe(center=center(c), radius=r_p).
    inc(M_S)
  end if

end for

References

[1] N. Akkiraju and H. Edelsbrunner. Triangulating the surface of a molecule. *Discrete Applied Mathematics*, 71:5–22, 1996.

[2] C. Anderson and M.D. Dahleh. Rapid computation of the discrete fourier transform. *SIAM J. Sci. Computing*, 17:913–919, 1996.

[3] C. Bajaj, H.Y. Lee, R. Merkert, and V. Pascucci. Nurbs based b-rep models for macromolecules and their properties. *Proceedings of the 4th Symposium on Solid Modeling and Applications (ACM Press)*, pages 217–228, 1997.

[4] C. Bajaj, V. Pascucci, A. Shamir, R. Holt, and A. Netravali. Dynamic maintenance and visualization of molecular surfaces. *Discrete Applied Mathematics*, 127:23–51, 2003.

[5] G. Beylkin. On the fast fourier transform of functions with singularities. *Appl. Comput. Harmon. Anal.*, 2:363–381, 1995.

[6] James F. Blinn. A generalization of algebraic surface drawing. *ACM Transactions on Graphics*, 1(3):235–256, July 1982.

[7] S. Boys. Electronic wave functions. i, a general method of calculation for the stationary states of any molecular system. *Proceedings of the Royal Society of London, Series A, Mathematical and Physical Sciences*, 200(1063):542–554, 1950.

[8] R. Carbo, L. Leyda, and M. Arnau. How similar is a molecule to another? an electron density measure of similarity between two molecular structures. *Int J Quantum Chem*, 17(6):1185–1189, 1980.

[9] R. Chen, L. Li, and Z. Weng. Zdock: an initial-stage protein-docking algorithm. *Proteins*, 52(1):80–87, 2003.

[10] R. Chen and Z. Weng. Docking unbound proteins using shape complementarity, desolvation, and electrostatics. *Proteins*, 47(3):281–294, 2002.

[11] M.L. Connolly. Analytical molecular surface calculation. *Journal of Applied Crystallography*, 16:548–558, 1983.

[12] M. Davis and J. McCammon. Electrostatics in biomolecular structure and dynamics. *Chem. Rev.*, 90:509–521, 1990.

[13] Bruce S. Duncan and Arthur J. Olson. Approximation and characterization of molecular surfaces. *Biopolymers*, 33:219–229, 1993.

[14] A. Dutt and V. Rokhlin. Fast fourier transform for nonequispaced data. *SIAM J. Sci. Computing*, 14:1368–1393, 1993.

[15] A. Dutt and V. Rokhlin. Fast fourier transform for nonequispaced data ii. *Appl. Comput. Harmon. Anal.*, 2:85–100, 1995.

[16] B. Elbel. Fast fourier transform for non equispaced data. *Approximation theory, C.K. Chui and L.L. Schumaker (eds.), Vanderbilt University Press*, 1998.

[17] D. Fischer, R.Norel, R.Nussinov, and H.J.Wolfson. 3-d docking of protein molecules. In *Proc. 4th Symp. On Combinatorial Pattern Matching*, pages 20–34. Springer Verlag, 1993.

[18] Matteo Frigo and Steven G. Johnson. FFTW: An adaptive software architecture for the FFT. In *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, volume 3, pages 1381–1384, Seattle, WA, May 1998.

[19] H.A. Gabb, R.M. Jackson, and M.J.E. Sternberg. Modelling protein docking using shape complementarity, electrostatics and biochemical information. *Journal of Molecular Biology*, 272(1):106–120, 1997.

[20] J. Grant and B. Pickup. A gaussian description of molecular shapes. *J. Phys. Chem.*, 99:3503–3510, 1995.

[21] E.E. Hodgkin and W.G Richards. Molecular similarity based on electrostatic potential and electric field. 1987.

[22] B. Honig and A. Nicholls. Classical electrostatics in biology and chemistry. *Science*, 268(1144-1149), 1995.

[23] E. Katchalski-Katzir, I. Shariv, M. Eisenstein, A. A. Friesem, C. Aflalo, and I. A. Vakser. Molecular surface recognition: Determination of geometric fit between proteins and their ligands by correlation techniques. *Proc. Natl. Acad. Sci. (USA)*, 89:2195–2199, 1992.

[24] J. A. Kovacs, Pablo Chac´on, Yao Cong, Essam Metwally, and Willy Wriggers. Fast rotational matching of rigid bodies by fast fourier transform acceleration of five degrees of freedom. *Acta Cryst. D*, 59:1371–1376, 2003.

[25] B. Lee and F. M. Rachards. The interpretation of protein structures: Estimation of static accessibility. *J. Mol. Biol*, 55:379–400, 1971.

[26] L. Li, R. Chen, and Z. Weng. Rdock: refinement of rigid-body protein docking predictions. *Proteins*, 53(3):693–707, 2003.

[27] J. G. Mandell, V. A. Roberts, M. E. Pique, V. Kotlovyi, J. C. Mitchell, E. Nelson, I. Tsigelny, and L. F. Ten Eyck. Protein docking using continuum electrostatics and geometric fit. *Protein Engineering*, 14(2):105–13, 2001.

[28] Nelson L. Max and Elizabeth D. Getzoff. Spherical harmonic molecular surfaces. *IEEE Computer Graphics & Applications*, 8:42–50, 1988.

[29] N.L. Max. Computer representation of molecular surfaces. *IEEE Computer Graphics and Applications*, 3(5):21–29, 1983.

[30] J. Mestres, D.C. Rohrer, and G.M. Maggiora. Mimic: A molecular -field matching program. exploiting applicability if molecular similarity approaches. *Journal of Computational Chemistry*, 18(7):934–, 1997.

[31] P.G. Mezey. *Shape in Chemistry*. VCH, New York, 1993.

[32] Daniel Potts, G. Steidl, and M. Tasche. *Fast Fourier transforms for nonequispaced data: A tutorial in Modern Sampling Theory: Mathematics and Applications*, chapter 12, pages 249–274. 2000.

[33] D. Ritchie. *Parametric Protein Shape Recognition*. Ph.D. thesis, University of Aberdeen, 1998.

[34] D. Ritchie and G. J. L. Kemp. Protein docking using spherical polar fourier correlations. In *PROTEINS: Structure Function & Genetics*. John Wiley & Sons, 1999.

[35] M.F. Sanner, A.J. Olson, and J.C. Spehner. Fast and robust computation of molecular surfaces. In *Proc. 11th Annual ACM Symposium on Computational Geometry*, pages C6–C7, 1995.

[36] Michel F. Sanner, Arthur J. Olson, and Jean-Claude Spehner. Reduced surface: An efficient way to compute molecular surfaces. *Biopolymers*, 38:305–320, 1996.

[37] J.C. Slater. Atomic shielding constants. *Phys. Rev.*, 36:57–64, 1930.

[38] G. Steidl. A note on fast fourier transforms for nonequispaced grids. *Advances in Computational Mathematics*, 9:337–352, 1998.

[39] E. Suli and A. Ware. A spectral method of characteristics for hyperbolic problems. *SIAM J. Numer. Anal*, 28:423–445, 1991.

[40] A. Varshney. Hierarchical geometric approximations. *PhD thesis, Dept. of CS, U. of North Carolina, Chapel Hill, TR-050*, 1994.

[41] A. Varshney and Jr.F.P. Brooks. Fast analytical computation of richard's smooth molecular surface. *Proceedings of the Visualization Conference, G. M. Nielson and D. Bergeron (ed.)*, pages 300–307, 1993.

[42] R. Voorintholt, M.T. Kosters, and G. Vegter. A very fast program for visualizing protein surfaces, channels and cavities. *Journal on Molecular Graphics*, 7:243–245, 1989.

[43] Anthony Ware. Fast approximate fourier transforms for irregularly spaced data. *SIAM Rev.*, 40:838.