

A Fast Protein-Protein Docking Algorithm Using Series Expansion in Terms of Spherical Basis Functions

Kazuya Sumikoshi¹

sumi@is.s.u-tokyo.ac.jp

Tohru Terada³

tterada@iu.a.u-tokyo.ac.jp

Shugo Nakamura²

shugo@bi.a.u-tokyo.ac.jp

Kentaro Shimizu^{1,2}

shimizu@bi.a.u-tokyo.ac.jp

¹ Department of Computer Science, Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

² Department of Biotechnology, Graduate School of Agricultural and Life Sciences, The University of Tokyo, 1-1-1 Yayoi, Bunkyo-ku, Tokyo 113-8657, Japan

³ Agricultural Bioinformatics Research Unit, Graduate School of Agricultural and Life Sciences, The University of Tokyo, 1-1-1 Yayoi, Bunkyo-ku, Tokyo 113-8657, Japan

Abstract

We describe a fast protein-protein docking algorithm using a series expansion in terms of newly designed bases to efficiently search the entire six-dimensional conformational space of rigid body molecules. This algorithm is an ab initio docking algorithm designed to list candidates of putative conformations from a global conformational space for unbound docking. In our algorithm, a scoring function is constructed from terms that are the inner products of two scalar fields expressing individual molecules. The mapping from a molecule to a scalar field can be arbitrarily defined to express an energy term. Since this scoring scheme has the same expressiveness as that of a method using a fast Fourier transform (FFT), it has the flexibility to introduce various physico-chemical energies. Currently, we are using scalar fields that approximate desolvation free energy and steric hindrance energy. Fast calculation of the scoring function for each conformation of the six-dimensional search space is realized by expansion of the fields in terms of basis functions which are combinations of spherical harmonics and modified Legendre polynomials, and the use of only low-order terms, which carry most of the information on the scalar field. We have implemented this algorithm and evaluated the computation time and precision by using actual protein structure data of complexes and their monomers. This paper presents the results for six unbound cases and in all the cases we obtained at least one conformation close to the native structures (interface RMSD < 3.0 Å) within the top 1000 candidates with about 40 seconds of computation time using a single Pentium4 2.4 GHz CPU.

Keywords: unbound protein-protein docking, global search, spherical harmonics, Legendre polynomials

1 Introduction

Since many proteins bring out their biological functions by binding to a specific partner protein at a specific site, determining the structure of a given complex is one of the most important focuses for molecular biology researchers. However, determining the putative structure of a protein-protein complex remains more difficult than determining ones for monomer or protein-small molecule complexes. Determining them experimentally for all the complexes that researchers may want to analyze is far from feasible because of the high costs and technical difficulty. Thus, computational methods to predict the putative conformations of a complex are expected to be helpful tools for analysis of interactions between pairs of given proteins.

The problem for computational methods can be defined as to output the predicted atomic coordinates of a complex by using the atomic coordinates of two molecules as inputs. This is often called “the docking problem” [6]. The problem of using the coordinates of the two molecules obtained by splitting the complex structure as an input is called the *bound* docking problem, and the problem of using coordinates determined individually (i.e., not as a complex) is called the *unbound* docking problem. The former is an artificial invention and is simply used as a test case for the evaluation of a method. The latter is of course the genuine problem we have to address. As each molecule of a complex usually changes its structure when forming a complex, the latter problem is the much harder one. The general assumption of the docking problem is that we have only the coordinates of the monomers as a given input; i.e., it assumes no additional data such as hints to the binding sites. That means one has to search the entire conformational space of a complex. Of course, a method is considered superior if it can additionally make use of possible binding site information, which is sometimes available in advance of a prediction. The protein-protein docking problem is inherently computationally harder than the protein-small molecule docking problem because of its larger degree of freedom due to the larger number of atoms involved.

Many computational docking methods have been proposed, and they have been summarized in review articles [6, 12]. Typically, a computational docking process consists of two stages. The first stage is a global search within a conformational space that produces a list of putative conformations. It usually uses a rigid body model for expressing the molecules, i.e., it is usually confined to a six-dimensional space of rotation and translation, because of the vastness of the conformational space without that constraint. Even with the six-dimensional space, it is still a large space to search with modern computers. Thus, this stage usually consists of a coarse search with a huge amount of light-weight energy calculations. The second stage refines the candidate conformations obtained from the first stage and results in fewer and finer candidates. This stage usually consists of a re-ranking and/or structural refining process with a fewer amount of heavy-weight calculations. In this paper, we focus on a method for the first stage.

As for the first stage of protein-protein docking, the major methods can be classified into three categories: methods based on surface feature point matching, methods that globally search the conformational space by using a fast Fourier transform (FFT), and methods using energy minimization starting from multiple initial conformations.

The methods of the first category involve extracting discrete feature points of a protein surface, e.g. convexes or concaves, and matching the points of the two molecules to get geometrically fitting conformations [3, 8, 9]. The algorithms usually used for matching are the maximal clique algorithm or geometric hashing. These methods are relatively fast if the number of feature points is kept low. However, the precision as a first stage docking is usually lower than the methods of the other two categories and they are apt to miss the correct conformation if the point extraction is not adequate. Moreover, their scoring functions are basically based on geometry and have less flexibility than scorings of an FFT-based method.

The methods of the second category utilize FFTs, which enables an efficient search of a three-dimensional translational space. Most docking methods fall into this category and they include the methods of Katchalski-Katzir *et al.* [7], FTDock [5] and ZDOCK [1]. Their speed is usually not faster than the first type, but their flexibility in designing the scoring function is higher.

The third type aims to list more precise candidates. Although these methods usually impose constraints or simplifications, they tend to take much more time compared with methods of the first two categories. Thus, they usually have the nature of second stage docking in addition to first stage docking.

Recently, many FFT-based methods have been proposed, which seems to be because of their good balance between flexibility in designing scoring functions, and reasonable precision and computation time. However, when these methods are applied to a large protein, it usually takes several hours to a few days to get a relatively high precision result with a single modern CPU.

The research field of computational docking is still in developing phase. The set of target complexes that each method can predict well varies depending on the method. That is, there is no method which can list a near-native candidate within the top few in almost all cases. Therefore, new searching methods and/or new scoring functions are being sought.

In this paper, we propose a computationally efficient and precise first stage docking algorithm by using a series expansion in terms of our newly designed orthonormal basis functions based on spherical harmonics. We aim to develop a method with higher computational efficiency than FFT-based methods while retaining flexibility in designing scoring functions, and to achieve good performance with unbound docking predictions. These features will lead to the applicability of finer and more complex scoring functions and to the possibility for encompassing interaction analysis among multiple proteins. There are several methods using spherical harmonics to express molecules for computational docking [4, 10]. Among these methods, the approach of Ritchie and Kemp [10] with radial basis functions has been relatively successful and seems to be a promising computationally efficient method. However, as is reported in their paper, the precision of the molecule representation drastically deteriorates as r (the distance from the origin) increases because its radial basis functions decay exponentially as r increases. This method thus becomes difficult to apply to large molecules or irregularly shaped molecules far from spherical in shape. To ameliorate these points and to achieve better performance, we use a combination of spherical harmonics and modified Legendre polynomials as basis functions, which have no decay for r , and incorporate desolvation free energy, which has been reported to reflect the binding energy well [13] for the scoring function.

2 Methods

2.1 Expression of Interaction Energy

A scoring function that reflects the interaction energy between two given molecules is constructed in terms of the inner products of two scalar fields within our framework; i.e., we express each molecule with N_s scalar fields; $f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_{N_s}(\mathbf{x})$ for molecule A and $g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_{N_s}(\mathbf{x})$ for molecule B , and then evaluate the scoring function of each conformation with

$$E(\mathcal{T}^A, \mathcal{T}^B) \equiv \sum_{i=1}^{N_s} w_i \int f_i^{\mathcal{T}^A}(\mathbf{x}) g_i^{\mathcal{T}^B}(\mathbf{x}) d\mathbf{x}$$

where w_i denotes the weight for the i -th term and \mathcal{T}^X means a rotational and/or translational operation on a field and $f^{\mathcal{T}}(\mathbf{x})$ means the field generated by applying \mathcal{T} to $f(\mathbf{x})$. Thus, the docking problem results in finding \mathcal{T}^A and \mathcal{T}^B with which $E(\mathcal{T}^A, \mathcal{T}^B)$ gets minimized. Figure 1 illustrates this scoring scheme. A set of $(\mathcal{T}^A, \mathcal{T}^B)$ should be systematically created such that it forms a six-dimensional search space. The set we used is described in section 2.5. The scalar fields can be arbitrarily defined by users of this framework to express the appropriate binding energy. This form of the scoring function has equal expressiveness to those of FFT-based methods. At this time, we defined it such that it expresses a pseudo desolvation free energy and steric hindrance, which is described in Section 2.4.

2.2 Fast Energy Calculation by Using Series Expansion

Each term of the scoring function described above, i.e., inner product of two scalar fields, can be efficiently calculated by expanding the scalar field in a series of orthogonal basis functions. We used a combination of spherical harmonics and modified Legendre polynomials as the basis functions. The mechanism of the fast calculation is described in the following paragraphs.

First, the two given scalar fields $f(\mathbf{x}), g(\mathbf{x})$ are expanded in terms of real spherical harmonics

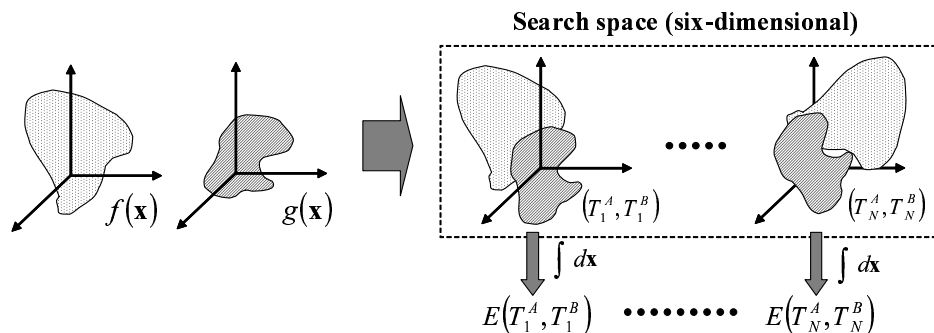


Figure 1: An illustration of the scoring and search scheme (in the case of one scalar field for each molecule). The given scalar fields $f(\mathbf{x})$ and $g(\mathbf{x})$ are transformed by $(T_1^A, T_1^B) \cdots (T_N^A, T_N^B)$, which form the six-dimensional search space. The scoring function for each conformation is calculated by integrating the product of the transformed fields.

$Y_{lm}^r(\theta, \phi)$ and modified Legendre polynomials $S_n(r)$. That is, expressing $f(\mathbf{x})$ with

$$f(\mathbf{x}) \approx \sum_{nlm}^{N_{max}, L_{max}} a_{nlm} S_n(r) Y_{lm}(\theta, \phi), \quad N_{max} \geq n \geq 0, L_{max} \geq l \geq |m| \geq 0$$

where a_{nlm} are the coefficients of the series, and N_{max} and L_{max} are the maximal orders of the expansions. The real spherical harmonics can be defined by using complex spherical harmonics:

$$Y_{lm}^r(\theta, \phi) = \begin{cases} \frac{(-1)^m}{\sqrt{2}} (Y_{lm}(\theta, \phi) + \bar{Y}_{lm}(\theta, \phi)), & m > 0 \\ Y_{l,0}(\theta, \phi), & m = 0 \\ \frac{(-1)^m}{\sqrt{2i}} (Y_{l,-m}(\theta, \phi) - \bar{Y}_{l,-m}(\theta, \phi)), & m < 0 \end{cases}$$

where $Y_{lm}(\theta, \phi)$ denotes a complex spherical harmonic, which are well-known orthogonal functions of mathematical physics, and $\bar{Y}_{lm}(\theta, \phi)$ denotes the complex conjugate of $Y_{lm}(\theta, \phi)$. The modified Legendre polynomials are defined as

$$S_n(r) \equiv \sqrt{\frac{6n+3}{a^3}} P_n\left(\frac{2}{a^3}r^3 - 1\right), \quad r \in [0, a]$$

where $P_n(x)$ is a Legendre polynomial and a is a parameter which determines the radius of the domain of the combined basis function. $S_n(r)$ has orthonormality as follows:

$$\int_0^a S_n(r) S_{n'}(r) r^2 dr = \delta_{nn'}$$

which is needed for ensuring the orthonormality of the combined basis functions:

$$\int S_n(r) Y_{lm}(\theta, \phi) S_{n'}(r) Y_{l'm'}(\theta, \phi) d\mathbf{x} = \delta_{nn'} \delta_{ll'} \delta_{mm'}$$

We hereafter refer to the combined basis function $B_{nlm}(\mathbf{x}) = B_{nlm}(r, \theta, \phi) \equiv S_n(r) Y_{lm}(\theta, \phi)$. This representation of $f(\mathbf{x})$ also means that we can express $f(\mathbf{x})$ with a_{nlm} by using these basis functions. The coefficients a_{nlm} can be calculated by taking the inner product of $f(\mathbf{x})$ and the corresponding basis function, i.e.,

$$a_{nlm} = \int f(\mathbf{x}) B_{nlm}(r, \theta, \phi) d\mathbf{x}$$

whose equality is derived from the orthogonality of the basis functions. Our method computes a_{nlm} by numerical integration using the discretized $f(\mathbf{x})$. Theoretically, we have to use an infinite number of coefficients to express $f(\mathbf{x})$ precisely. However, we can usually express $f(\mathbf{x})$ with reasonable precision by using only low-order terms because the $f(\mathbf{x})$ used for docking usually does not have many high-frequency vibrations. In other words, we can compress the information of $f(\mathbf{x})$ by using the representation with a_{nlm} .

By expanding fields, the inner product for calculating the scoring function can be reduced to the inner product of the two coefficient vectors, which needs far fewer computations. That is,

$$\begin{aligned} \int f(\mathbf{x})g(\mathbf{x})d\mathbf{x} &\approx \sum_{nlm} \sum_{n'l'm'} a_{nlm}b_{n'l'm'} \int B_{nlm}(\mathbf{x})B_{n'l'm'}(\mathbf{x})d\mathbf{x} \\ &= \sum_{nlm} \sum_{n'l'm'} a_{nlm}b_{n'l'm'}\delta_{nn'}\delta_{ll'}\delta_{mm'} = \sum_{nlm} a_{nlm}b_{nlm} \end{aligned}$$

Thus, the total scoring function $E(\mathcal{T}_1, \mathcal{T}_2)$ can be reduced to

$$\begin{aligned} E(\mathcal{T}^A, \mathcal{T}^B) &= \sum_{i=1}^{N_s} w_i \int f_i^{\mathcal{T}^A}(\mathbf{x})g_i^{\mathcal{T}^B}(\mathbf{x})d\mathbf{x} \\ &= \sum_{i=1}^{N_s} w_i \sum_{nlm} a_{i,nlm}^{\mathcal{T}^A} b_{i,nlm}^{\mathcal{T}^B} \end{aligned}$$

where $a_{i,nlm}^{\mathcal{T}^A}$ and $b_{i,nlm}^{\mathcal{T}^B}$ are the coefficients of the transformed scalar fields $f_i^{\mathcal{T}^A}(\mathbf{x})$ and $g_i^{\mathcal{T}^B}(\mathbf{x})$, respectively. $a_{i,nlm}^{\mathcal{T}^A}$ and $b_{i,nlm}^{\mathcal{T}^B}$ can be efficiently computed from the original coefficients, $a_{i,nlm}$ and $b_{i,nlm}$.

2.3 Fast Rotational and Translational Operations

As described above, we need to obtain the transformed coefficients, $a_{nlm}^{\mathcal{T}^A}$ and $b_{nlm}^{\mathcal{T}^B}$, for the conformational space search. The obvious way to obtain them is to recompute the discretized transformed field $f^{\mathcal{T}^A}(\mathbf{x})$ and then compute a numerical integration to get the coefficients $a_{nlm}^{\mathcal{T}^A}$. This method, however, requires many computational steps. In contrast, the method we will describe hence computes the transformed coefficients directly from the original coefficients and consequently does not need a costly numerical integration. Since the transformation consists of rotational and translational operations, we will describe an efficient algorithm for each operation.

First, considering the coefficients of a rotated field $a_{nlm}^{\mathcal{R}}$ where \mathcal{R} denotes a rotational operation to a field. $a_{nlm}^{\mathcal{R}}$ can be formulated as:

$$a_{nlm}^{\mathcal{R}} = \sum_{m'=-l}^l a_{nlm'} R_{mm'}^l(\mathcal{R}^{-1})$$

by using the fact that rotated real spherical harmonics can be described as

$$Y_{lm}(\theta', \phi') = \sum_{m'=-l}^l R_{m'm}^l(\mathcal{R}) Y_{lm'}(\theta, \phi)$$

where θ' and ϕ' denote the angular coordinates rotated with \mathcal{R} and $R_{m'm}^l(\mathcal{R})$ denotes the rotation matrix for real spherical harmonics [2] determined by \mathcal{R} .

Next, we will address the translational operation along the z -axis. The reason for limiting the translation to only along the z -axis is that we can construct the six-dimensional search by combining

just a rotational operation and a translational operation along the z -axis (as described later). The coefficients of the translated field $a_{nlm}^{S_{\Delta z}}$ where $S_{\Delta z}$ denotes a translational operation of $(0, 0, \Delta z)$ on a field can be obtained by

$$\begin{aligned} a_{nlm}^{S_{\Delta z}} &= \sum_{n'l'} a_{n'l'm} \int_0^a \int_0^\pi S_{n'}(r'') S_n(r) P_{l'}^{|m|}(\cos \theta'') P_l^{|m|}(\cos \theta) r^2 \sin \theta d\theta dr \\ &\equiv \sum_{n'l'} a_{n'l'm} O_{n',n,l,|m|}(\Delta z) \end{aligned}$$

where r'' and θ'' denote the spherical coordinates translated by S_{dz} and $P_l^m(t)$ denotes a normalized Legendre polynomial and $O_{n',n,l,|m|}(\Delta z)$ denotes the overlapping integral.

The values of $O_{n',n,l,|m|}(\Delta z)$ are calculated in advance by numerical integrations, and their results are stored in a table for later uses because they are independent of the contents of the scalar fields. As one can see from the double summations, this operation costs more than the rotational operation.

2.4 Scoring Functions

As stated above, we can use an arbitrary scoring function if it consists of terms of inner products between two scalar fields. That is, designing the scoring function means designing the mapping from a given molecule to a scalar field. Currently, we are using a scoring function designed to approximate the difference in desolvation free energy and steric hindrance energy between the free and the complex form of the two molecules.

The desolvation free energy is the energy required to move atoms within water to the interior of molecules. This energy can be roughly estimated by using the atomic contact energy (ACE) [13]. ACE is calculated by counting the number of contacting atoms, which is classified into 18 types, and it is reported that the calculated energy is very close to the experimental result for the protein binding free energy. Based on these results, we have designed a mapping function which reflects the contact between atoms. Following this method, the difference between free and complex forms can be calculated by summing up the energy $E_{i,j}$ for each pair of atoms (whose atom type is i and j respectively) which belong to different molecules and the distance between which is lower than a certain threshold (6.0 Å in the paper [13]). This energy can be expressed by using 18 scalar fields for each molecule. Denoting the scalar field for molecules A and B as ρ_i^A and ρ_i^B ($i = 1, \dots, 18$), a definition of these scalar fields that will lead to the expression of ACE is:

$$\begin{aligned} \rho_i^A(\mathbf{x}) &= n_{atoms}^A(i, \mathbf{x}; d_{thr}) \\ \rho_i^B(\mathbf{x}) &= \begin{cases} 1 & \text{if an atom of type } i \text{ exists at } \mathbf{x} \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where $n_{atoms}^A(i, \mathbf{x}; d_{thr})$ denotes the number of atoms of type i in molecule A whose distance from \mathbf{x} is equal to or less than a pre-defined threshold d_{thr} . The energy can be calculated using this field definition:

$$E = \sum_{i,j}^{18} E_{i,j} \int \rho_i^A(\mathbf{x}) \rho_j^B(\mathbf{x}) d\mathbf{x}$$

where $E_{i,j}$ is the statistically obtained energy used in ACE. Although this calculation fits our framework, there are two problems: (1) Since $\rho_i^B(\mathbf{x})$ consists of delta functions, the coefficients of its series expansion will not diminish as the order increases. That is, the precision loss caused by picking only the low-order terms becomes high. (2) It requires 18^2 inner product calculations, which costs too much.

To solve the first problem, we have used an alternative field definition that also reflects atomic contacts:

$$\begin{aligned}\rho_i^A(\mathbf{x}) &= n_{atoms}^A(i, \mathbf{x}; d'_{thr}) \\ \rho_i^B(\mathbf{x}) &= n_{atoms}^B(i, \mathbf{x}; d'_{thr})\end{aligned}$$

where d'_{thr} is a pre-defined threshold slightly larger than $d_{thr}/2$ (4.0 Å in our experiment). With this definition, the energy is dependent on the distance between atoms, unlike in the case of the original definition.

To solve the second problem, we have reduced the number of terms by using matrix diagonalization. The energy is described as $E = \int \sum_{i,j}^{18} E_{i,j} \rho_i^A(\mathbf{x}) \rho_j^B(\mathbf{x}) d\mathbf{x}$. The integrand can be expressed using matrices:

$$\sum_{i,j}^{18} E_{i,j} \rho_i^A(\mathbf{x}) \rho_j^B(\mathbf{x}) = \sum_{i,j}^{18} E_{i,j} p_i p_j \rho_i^{A'}(\mathbf{x}) \rho_j^{B'}(\mathbf{x}) = (\rho'_A)^T \mathbf{E}' \rho'_B$$

where p_i is the statistically obtained frequency of atoms of type i , $\rho_i^{X'}(\mathbf{x}) \equiv \rho_i^X(\mathbf{x})/p_i$, $\rho'_X \equiv (\rho_1^{X'}(\mathbf{x}), \rho_2^{X'}(\mathbf{x}), \dots, \rho_{18}^{X'}(\mathbf{x}))^T$ and $\mathbf{E}' \equiv \begin{pmatrix} p_1 p_1 E_{1,1} & \cdots & p_1 p_{18} E_{1,18} \\ \vdots & \ddots & \vdots \\ p_{18} p_1 E_{18,1} & \cdots & p_{18} p_{18} E_{18,18} \end{pmatrix}$. The frequencies p_i

are used to normalize the ranges of $\rho_i^X(\mathbf{x})$ for the convenience of picking terms, described in the end of this paragraph. Since \mathbf{E}' is symmetric because $E_{i,j} = E_{j,i}$, it can be diagonalized using orthogonal matrices.

$$\begin{aligned}(\rho'_A)^T \mathbf{E}' \rho'_B &= (\rho'_A)^T \mathbf{U} \mathbf{\Omega} \mathbf{U}^T \rho'_B = (\eta_A)^T \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_{18} \end{pmatrix} \eta_B \\ &= \sum_i^{18} \lambda_i \eta_i^A(\mathbf{x}) \eta_i^B(\mathbf{x})\end{aligned}$$

where \mathbf{U} and $\mathbf{\Omega}$ are respectively orthogonal and diagonal matrices determined from \mathbf{E}' , and $\eta_X \equiv \mathbf{U}^T \rho'_X$. Using this transformation, E can be rewritten as

$$\begin{aligned}E &= \int \sum_i^{18} \lambda_i \eta_i^A(\mathbf{x}) \eta_i^B(\mathbf{x}) d\mathbf{x} \\ &= \sum_i^{18} \lambda_i \int \eta_i^A(\mathbf{x}) \eta_i^B(\mathbf{x}) d\mathbf{x}\end{aligned}$$

which means 18^2 inner product calculations get reduced to 18 inner product calculations. Furthermore, we can reduce the calculation by picking only the terms which contribute largely, i.e., terms which have large λ_i . Our experiment used the top two terms. This term reduction scheme can be applied not only to ACE but also to any energy of the form $E = \sum_{i,j}^N E_{i,j} \int \rho_i^A(\mathbf{x}) \rho_j^B(\mathbf{x}) d\mathbf{x}$ under the condition $E_{i,j} = E_{j,i}$.

Since the terms described above do not take into account repulsive effects between atoms, we introduced a term for steric hindrance to exclude heavily overlapping conformations. The definition of the field is as follows:

$$\rho^X(\mathbf{x}) = \sum_i^{N_{atom}} w_{sc}(i) I_{vdW}(i, \mathbf{x})$$

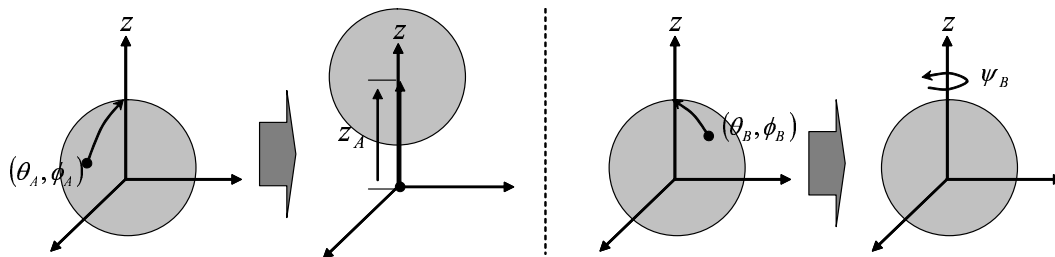


Figure 2: A conformation specified with $(\theta_A, \phi_A, z_A, \theta_B, \phi_B, \psi_B)$. The left figure illustrates rotational and translational operations for molecule A. The first rotation is an operation with which a unit vector determined by (θ_A, ϕ_A) is rotated to be parallel to the z -axis. The next operation is a translation along the z -axis. The right figure illustrates rotational operations for molecule B. The first rotational operation is performed in the same manner as the first operation for molecule A. The second rotational operation is a rotation around the z -axis.

where $w_{sc}(i)$ is a weight value determined by whether atom i is a surface atom or core atom, and $I_{vdW}(i, \mathbf{x})$ is 1 if the distance between the center of atom i and point \mathbf{x} is less than or equal to the vdW radius of atom i , and 0, otherwise. Basically, repulsion takes effect within the vdW radius of an atom. However, to compensate for structural changes induced by forming a complex, the weights and the vdW radii of surface atoms are set smaller than those of core atoms. In addition, the vdW radii of core atoms are set larger than the default vdW radii to fill the space not occupied by atoms with the default vdW radii. An atom is classified as a surface atom if its solvent-accessible surface area is more than 1.0 \AA^2 . Otherwise, it is classified as a core atom.

2.5 Outline of the Docking Procedure

We have described the basic principles of our method. We will now describe the overall docking procedures, including precalculations, preprocessings, and the actual process of docking computation.

The precalculation that is required before the docking computation is the creation of the look-up table for the overlapping integrals $O_{n',n,l,|m|}(\Delta z)$. This table is created for typical parameters of the maximal orders of expansions and steps of Δz . This calculation is required only once for a system, unless we have to perform docking with irregular parameters with which no tables have been created.

Next, the series expansion is required for each protein. The expansion is performed by a numerical integration as described in Section 2.2 with the centroid of the molecules moved to the origin. This process is required only once for a molecule, provided both of the coefficients of the target molecules are expanded using the same basis functions, i.e., using the same parameter a for the radial basis function $S_n(r)$.

The actual docking process is performed by evaluating the energy of each conformation created from the search space of five rotational dimensions and one translational dimensions. This search space decomposition is equivalent to that of Ritchie *et al.* [10] A conformation is specified with $(\theta_A, \phi_A, z_A, \theta_B, \phi_B, \psi_B)$ where (θ_A, ϕ_A) determines the rotational operation on the field of the molecule A, z_A determines the translational operation along the z -axis on that rotated field of the molecule A and $(\theta_B, \phi_B, \psi_B)$ determines the rotational operation on the field of the molecule B. Figure 2 illustrates these operations for six-dimensional search. The basic procedure is described in Algorithm 1 in pseudo code.

There are four points to note about Algorithm 1: (1) The ‘‘almost’’ evenly distributed points on a unit sphere can be obtained in various ways [11]. We are currently using recursive subdivision of a unit octahedron or icosahedron. (2) ‘‘The rotational operation with which p_i comes on the z -axis’’ has one freedom left. Any appropriate function will do as long as it uniquely maps a point p on a unit sphere

The parameters used throughout this experiment were as follows:

- maximal order of coefficients: $N_{max} = 8$, $L_{max} = 10$
- radius of the domain of the basis function: 35.0 Å
- granularity of search:
 - $\Delta\theta_A, \Delta\phi_A, \Delta\theta_B, \Delta\phi_B$: approx. 15.8 deg. (derived from recursive subdivision of an icosahedron)
 - $\Delta\psi_B$: 15.0 deg.
 - Δz_A : 1.0 Å
- factor for vdW radii of surface and core: 0.9 for surface, 1.2 for core
- weights $w_{sc}(i)$ for the field of steric hindrance: 0.2 for surface, 1.0 for core
- weights w_i used for each term: 1.0 for desolvation free energy, 120.0 for steric hindrance

All of the molecules listed in Table 1 fit in a sphere of radius 35.0 Å.

3.1 Computation Time

The computation time required for docking was about 40 seconds with a single 2.4 GHz Pentium4 processor. Although the computation time depends on the sum of the radii of two molecules and their shapes, the difference between cases was merely on the order of seconds. The total number of conformations generated and evaluated within the computation was on the order of 10^7 to 10^8 .

To make a comparison with FFT-based methods, we ran the freely available FTDock [5] program (with `fftw-2.1` as an FFT library) in the same computational environment. 1AKZ and 1UGI(A) were used as input data. With the default parameters of FTDock (angle step: 12 deg., translation step: 0.7 Å, electrostatics: on), FTDock took about 18 hours. With search parameters close to our parameters (angle step: 15 deg., translation step: 1.0 Å, electrostatics: on), it took about 100 minutes. Although a direct comparison is difficult, we can say that our method was about 170 times (to 1800 times) faster with nearly the same level of precision, based on the results shown in the paper on FTDock (compare our results shown in Table 3 and 4 in Section 3.2 with “Before filtering” column of Table 2 in their paper [5]). This speed-up was due to the several advantages of our method: (1) The rotational operations to the coefficients, which are required for a five-dimensional search, can be performed very quickly by directly manipulating the original coefficients. In FFT-based methods, these operations involve a rotational operation in a real space and application of FFT to that rotated space. (2) The amount of memory required to express a molecule is less than in FFT-based methods, which leads to faster computations by making it possible for many intermediate results to be cached. In FFT-based methods, the data required for a molecule tends to be larger than that of ours because of the coefficients for a molecule have to cover an area large enough to contain both target molecules and because of the restriction that the grid spacing for expressing a molecule and the translational step of the search must be equal. In contrast, the coefficients of our method only have to cover their own molecule and the search step can be chosen arbitrarily. The ratio of the required memory (FTDock/Ours) for a field with the default parameters is $(128^3) / (9 \times 11^2) \approx 1925.8$. (3) With our method, it is easy to selectively search only a part of the space; e.g., we can exclude conformations in which two molecules completely overlap or two molecules are too far apart to interact. FFT-based methods do not have such selectivity because the three-dimensional translational search results have to be calculated all at once.

Table 2: Best interface RMSD in the top rank * (Total number of candidates: about 10^7 to 10^8).

complex	best I-RMSD in 8000 (rank)		best I-RMSD in 1000 (rank)		best I-RMSD in 100 (rank)		best I-RMSD in 10 (rank)	
1UGH	1.71	(692)	1.71	(692)	2.02	(10)	2.02	(10)
1BRB	1.43	(5479)	2.07	(598)	2.84	(71)	4.55	(9)
2SIC	1.76	(4981)	2.95	(377)	4.68	(46)	10.64	(2)
2PTC	0.99	(2647)	2.25	(80)	2.26	(80)	7.20	(3)
1CHO	1.49	(3961)	2.53	(811)	6.06	(55)	6.23	(4)
1CGI	1.76	(7909)	2.96	(517)	4.22	(100)	5.32	(1)

* RMSD is in Å

Table 3: Rank of the first near-native and its interface RMSD in the top rank.

complex	rank of 1st hit <2.5Å(I- RMSD)		rank of 1st hit <3.0Å(I- RMSD)		rank of 1st hit <4.0Å(I- RMSD)		rank of 1st hit <5.0Å(I- RMSD)	
1UGH	10	(2.02)	10	(2.02)	10	(2.02)	4	(4.67)
1BRB	598	(2.07)	21	(2.84)	15	(3.96)	2	(4.92)
2SIC	4981	(1.76)	377	(2.96)	377	(2.96)	17	(4.99)
2PTC	80	(2.26)	80	(2.26)	62	(3.33)	11	(4.75)
1CHO	1357	(2.11)	811	(2.53)	232	(3.75)	232	(3.75)
1CGI	6712	(2.25)	517	(2.96)	229	(3.57)	53	(4.71)

3.2 Qualities of the Predicted Structures

We measured the quality of the obtained candidate complex structure by using the interface RMSD. The interface RMSD is the RMSD of C_α atoms in interface residues, which are residues that have at least one atom within 10 Å of any atom belonging to the other molecule. We used a interface RMSD threshold of 2.5 Å as a first criteria and 3.0 Å as a second criteria for the near-native structures.

Table 2 shows the best interface RMSD within a certain number of candidates. Table 3 and Table 4 show the rank of the first near-native structure and the number of the near-natives within the top 4000 for various thresholds. Table 2 indicates that our program lists at least one near-native (≤ 2.5 Å) structure within the top 8000 and at least one near-native (≤ 3.0 Å) structure within the top 1000. This means that a good measure is to use top 8000 or 1000 candidates as the inputs to the next refinement process, what we call “second stage docking”.

We also investigated the candidate structures and found several tendencies for the positions of molecules. The 3-d graphs plotted with the positions of the centroids of the ligand (the smaller molecule) viewed from the receptor (the larger molecule) for each conformation ranked within the top 50 (graph not shown) show that the plotted positions form one to a few clusters and at least one of the clusters, usually the largest one, is located close to the position of the native structure for all the cases. This means that this method has the large potential for reliably estimating the probable binding sites of receptors.

We also created the graphs of the positions of the centroids of the receptor viewed from the ligand. In the cases of the targets whose best interface RMSD in the top 10 was below 5.0 Å, i.e., 1UGH and 1BRB, the both graphs show similar tendency. However, those graphs of the targets whose best interface RMSD in the top 10 was above 5.0 Å, almost all the clusters tend to be far from the point

Table 4: Number of near-natives within the top 4000.

complex	# of hits <2.5Å	# of hits <3.0Å	# of hits <4.0Å	# of hits <5.0Å
1UGH	7	13	24	50
1BRB	7	18	48	89
2SIC	0	2	7	19
2PTC	14	33	79	168
1CHO	16	25	46	73
1CGI	0	2	35	144

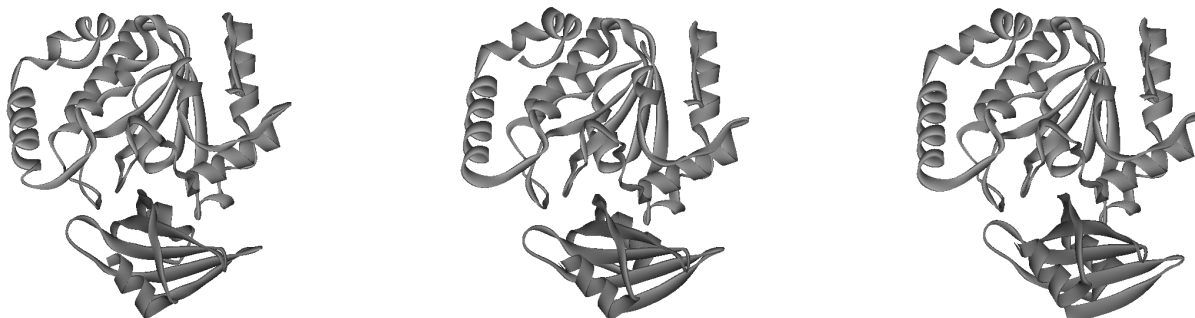


Figure 3: The predicted structure with the best interface RMSD within the top 1000 and the first near-native structure (I-RMSD < 2.5 Å) for 1AKZ and 1UGI(A) (unbound docking). (left: the native structure (1UGH), center: the predicted structure with the best I-RMSD (interface RMSD: 1.71 Å, rank: 692), right: the first near-native structure (interface RMSD: 2.02 Å, rank: 10))

of the native structure and this is one of the major factors of having the bad interface RMSD in the top 10 or 100.

Figure 3 shows some samples of our results for unbound docking. They are the candidate structure with the best interface RMSD within the top 1000 and the first near-native structure for 1AKZ and 1UGI(A).

4 Conclusion

We proposed a new protein-protein docking algorithm using a series expansion of scalar fields in terms of newly designed basis functions for fast computations. Our computational framework has the same level of flexibility for designing the scoring function as the FFT-based methods. Our algorithm focused on the initial stage of unbound docking, and aimed to show reasonable performance even without incorporating knowledge of the binding site, although it can easily take such knowledge into account by introducing constraints into the search range. Our preliminary computational results using a scoring function based on desolvation free energy and steric hindrance showed that our program could output the predicted structures comparable in quality to the predictions of FFT-based docking methods without incorporating binding site information and could complete the task in less than a minute using a single 2.4 GHz Pentium4 processor, which is about 170 times (to 1800 times) faster than FFT-based methods. This speed-up will give more time to the second stage of docking to refine the results and will result in much better overall speed and quality of the prediction.

For dealing with very large proteins, we can express the molecule with using larger radius a for basis functions or using multiple spheres of radius a whose union covers an entire molecule. We have

developed the latter method and its evaluation will be reported in another paper.

Although the preliminary results are reasonably good for the initial stage of docking, we can further improve the performance by tuning the parameters and scoring function, including the introduction of the terms for electrostatics. We are also planning to implement a coarse-to-fine hierarchical search and make our own refinement stage docking algorithm.

Acknowledgments

This research is supported by a Grant-in-Aid for Scientific Research on Priority Areas "Genome Information Science" from the Ministry of Education, Science and Culture of Japan.

References

- [1] Chen, R. and Weng, Z., Docking unbound proteins using shape complementarity, desolvation, and electrostatics, *Proteins*, 47(3):281–294, 2002.
- [2] Choi, C. H., Ivanic, J., Gordon, M. S., and Ruedenberg, K., Rapid and stable determination of rotation matrices between spherical harmonics by direct recursion, *J. Chem. Phys.*, 111:8825–8831, 1999.
- [3] Connolly, M. L., Shape complementarity at the hemoglobin alpha 1 beta 1 subunit interface, *Biopolymers*, 25(7):1229–1247, 1986.
- [4] Duncan, B. S. and Olson, A. J., Applications of evolutionary programming for the prediction of protein-protein interactions, *Evolutionary programming V : proceedings of the Fifth Annual Conference on Evolutionary Programming*, 411–417, 1996.
- [5] Gabb, H. A., Jackson, R. M., and Sternberg, M. J., Modelling protein docking using shape complementarity, electrostatics and biochemical information, *J. Mol. Biol.*, 272(1):106–120, 1997.
- [6] Halperin, I., Ma, B., Wolfson, H., and Nussinov, R., Principles of docking: An overview of search algorithms and a guide to scoring functions, *Proteins*, 47(4):409–443, 2002.
- [7] Katchalski-Katzir, E., Shariv, I., Eisenstein, M., Friesem, A. A., Aflalo, C., and Vakser, I. A., Molecular surface recognition: Determination of geometric fit between proteins and their ligands by correlation techniques, *Proc. Natl. Acad. Sci. USA*, 89(6):2195–2199, 1992.
- [8] Kuntz, I. D., Blaney, J. M., Oatley, S. J., Langridge, R., and Ferrin, T. E., A geometric approach to macromolecule-ligand interactions, *J. Mol. Biol.*, 161(2):269–288, 1982.
- [9] Norel, R., Petrey, D., Wolfson, H. J., and Nussinov, R., Examination of shape complementarity in docking of unbound proteins, *Proteins*, 36(3):307–317, 1999.
- [10] Ritchie, D. W. and Kemp, G. J., Protein docking using spherical polar Fourier correlations, *Proteins*, 39(2):178–194, 2000.
- [11] Saff, E. B. and Kuijlaars, A. B. J., Distributing many points on a sphere, *Mathematical Intelligencer*, 19(1):5–11, 1997.
- [12] Smith, G. R. and Sternberg, M. J., Prediction of protein-protein interactions by docking methods, *Curr. Opin. Struct. Biol.*, 12(1):28–35, 2002.
- [13] Zhang, C., Vasmatzis, G., Cornette, J. L., and DeLisi, C., Determination of atomic desolvation energies from the structures of crystallized proteins, *J. Mol. Biol.*, 267(3):707–726, 1997.