2011 International Conference on Bioscience, Biochemistry and Bioinformatics
IPCBEE vol.5 (2011) © (2011) IACSIT Press, Singapore

Protein-Protein Docking Using Multi-layered Spherical Basis Functions

Kazuya Sumikoshi, Tohru Terada, Shugo Nakamura, Kentaro Shimizu
Graduate School of Agricultural and Life Sciences
The University of Tokyo
1-1-1 Yayoi, Bunkyo-ku, Tokyo 113-8657, Japan

Abstract—**The prediction of the structure of protein-protein complexes is an important but difficult problem mainly due to the huge computational cost required to search a configuration space. In this paper, we propose a fast global search docking algorithm that uses our novel multi-layered spherical basis functions (MLSBFs) to efficiently search the six-dimensional space of a rigid-body model. In our approach, the three-dimensional space is divided into layers and the scalar fields in each layer, used for calculating scores, are expanded by the MLSBFs to get the coefficients that express the fields. The MLSBFs enable the efficient usage of coefficients to express fields and thus an efficient global search. Our experiments showed that our program was able to generate predicted structures within a few minutes using a single 2.4 GHz Pentium 4 processor, and succeeded in listing at least one near-native structure, whose interface RMSD is less than 2.5 angstrom, within the top 100 in 5 out of 7 cases.**

Keywords-component; protein-protein docking; spherical harmonics

I. Introduction

Protein-protein interactions play major roles in biological functions. In understanding the detailed mechanisms of protein interactions, it is indispensable to have knowledge about the structure of a target complex. Although there are experimental techniques, such as X-ray crystallography or NMR, to obtain the structure of a complex, determining all the structures of interest with them is infeasible due to the cost and technical difficulties. Therefore, computational techniques to predict the structure of a protein-protein complex, known as protein-protein docking, have been a focus of attention since they may be able to provide the structural information that experimental methods cannot provide.

Protein-protein docking can be formally described as a process of predicting the three-dimensional structure of a complex from the known structures of the unbound monomers. It is usually assumed that all the information available in a prediction process is the coordinates of the monomers; i.e., without any additional data such as hints to the binding sites. A number of docking methods have been proposed, and they are comprehensively summarized in review articles [1], [2], [3]. The main difficulty of this problem is the huge number of degrees of freedom the system inherently has. Thus we have to adopt some approaches that can cope with the vastness of the search space in order to reduce the computation time. The typical strategy to handle the problem is to divide the process into two stages. The first stage (initial stage) performs global search and generates probable candidates. It usually uses simplified energy (or score) terms and a rigid body model, i.e., intramolecular flexibility is neglected, to considerably reduce the search space to six-dimensional relative orientation space. The second stage (refinement stage) refines the generated candidates by using more computationally expensive calculations such as re-ranking and/or structural refining processes. This stage usually corresponds to performing finer search around the candidates. Obviously, achieving good performance in the initial stage is crucial for the whole process to be successful.

The methods for the initial stage proposed so far ranges from ones based on surface feature point matching [4], [5], [6], [7], [8], ones using energy minimization starting from multiple initial conformations [9], [10], [11], and ones that globally and uniformly search the space by using a fast Fourier transform (FFT) [12], [13], [14]. As with any other algorithm, the methods described above have trade-offs between computation time and accuracy. However, many of the methods proposed recently use FFTs. This seems to be because of their good balance between flexibility in designing scoring functions, and reasonable accuracy and computation time. The methods in this category can efficiently search the translational space by representing an interaction energy function in terms of the inner products of the scalar fields generated from each molecule, and performing fast computation of the energy functions for each translation by using FFTs. However, the speed of FFT-based methods is not fast enough not to need any further speed-ups. When these methods are applied to a large protein, it usually takes several hours to a few days to get a result with relatively high accuracy by using a single modern CPU. Therefore, further improvement in efficiency is of great use.

One of the approaches for speed-up is to add some ingenuity to rotational operations, which can take up five out of six-dimensional search space (cf. Fig. 2), and which conventional FFT-based techniques cannot exploit. Expressing an object or a field with coefficients calculated by a series expansion in terms of spherical harmonics is known to be convenient for rotational operations. Some approaches use spherical harmonics for protein-protein docking [15], [16], [17], and among these methods, the approach by Ritchie [16], [17] that also uses radial basis functions is reported to be fairly successful in speeding up the docking process, especially for searching the rotational space, and seems to be a promising method. However, as is reported in their paper, the precision of expressing a field drastically deteriorates as r (the distance from the origin) increases because its radial basis functions decay exponentially as r increases. It means that it is difficult to apply the method to large molecules. To ameliorate these points, the authors have proposed a method in the past that uses a combination of spherical harmonics and modified Legendre polynomials as basis functions, which have no decay for r [18]. Here, we further extended, or generalized, the method in [18] by dividing r space into multiple regions, and designed radial basis functions tailored for each region. To put it another way, $\mathbf{R}^3$ space is divided into layers and scalar fields in each layer are expanded in terms of the basis functions which are composed of spherical harmonics and the radial basis functions for the corresponding layer. This approach will allow us to express a field more efficiently and flexibly because we can finely adjust the number of coefficients to use for each layer depending on the importance of the layer; usually a layer that contains the region of a molecular surface is more important for calculating interaction energies than a layer containing only an internal core of a molecule. Since the number of coefficients directly affects the computation time, efficient use of coefficients proposed in this paper leads to faster calculation while retaining a fine representation of fields.

II. Methods

A. Form of Scoring Functions

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

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

where w_i denotes the weight for the i-th term, T^X means a rotational and/or translational operation on a field for molecule X and $f^{\mathrm{T}}(\mathbf{x})$ means the field generated by applying T to $f(\mathbf{x})$. Thus, the docking problem results in finding T^A and T^B with which $E\left(\mathrm{T}^A, \mathrm{T}^B\right)$ is minimized.

These scalar fields can be arbitrarily defined by users of this framework to express an appropriate binding energy. This form of a scoring function has equal expressiveness to those of FFT-based methods.

B. Basis Functions Used for Expanding Scalar Fields

The fast score computation is enabled by expanding scalar fields in terms of orthogonal basis functions. The basis functions that we use for $\mathbf{R}^3$ can be described as

$$B_{k,n,l,m}(\mathbf{x}) = B_{k,n,l,m}(r,\theta,\phi) \equiv S_{k,n}(r) Y_{l,m}(\theta,\phi),$$

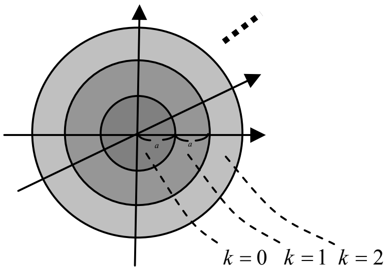

Figure 1. An illustration of the regions of the multi-layered spherical basis functions (MLSBFs). The region for each k denotes the area where the basis functions for k have non-zero values.

where $S_{k,n}(r)$ denotes the radial part of the basis functions that we designed, and $Y_{l,m}(\theta,\phi)$ denotes the normalized real spherical harmonics used for the angular part of the basis functions. The normalized real spherical harmonics have the orthonormality of

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

As for the basis functions for the radial part, r space is first divided into multiple intervals I_k of widths a, i.e., $I_k \equiv [ka,(k+1)a)$ for $k = 0,1,\ldots$, and then $S_{k,n}(r)$ for each region are defined such that $S_{k,n}(r) = 0$ if $r \notin [ka,(k+1)a)$ and $\int_0^{\infty} S_{k,n}(r) S_{k,n'}(r) r^2 dr = \delta_{nn'}$. We derived the set of functions that fulfill the specified conditions by using the Gram-Schmidt process. More precisely, we get

$$S_{k,n}(r) \equiv \begin{cases} \sqrt{\dfrac{8}{N_{k,n}^2 a^3}} h_{k,n}\left(\dfrac{2}{a} r - 2k - 1\right), & r \in [ka,(k+1)a) \\ 0, & \text{otherwise} \end{cases}$$

where $h_{k,n}(x)$ denote the orthogonal polynomials built by using the Gram-Schmidt process with weight function $(x+2k+1)^2$ and interval $[-1,1]$, and $N_{k,n}$ is the norm of $h_{k,n}(x)$. That is, $h_{k,n}(x)$ are derived by using the recurrence relation

$$h_{k,i+1}(x) = \left[x - \frac{\langle x h_{k,i} \mid h_{k,i} \rangle}{\langle h_{k,i} \mid h_{k,i} \rangle} \right] h_{k,i}(x) - \left[\frac{\langle h_{k,i} \mid h_{k,i} \rangle}{\langle h_{k,i-1} \mid h_{k,i-1} \rangle} \right] h_{k,i-1}(x)$$

with $h_{k,0}(x) = 1$, $h_{k,-1}(x) = 0$, and $\langle f_i \mid f_j \rangle \equiv \int_{-1}^{1} f_i(x) f_j(x) (x+2k+1)^2 dx$. Then we can see that they have the orthonormality of $\int_0^{\infty} S_{k,n}(r) S_{k',n'}(r) r^2 dr = \delta_{kk'} \delta_{nn'}$. Thus, the combined basis functions have the orthonormality of $\int S_{k,n}(r) Y_{l,m}(\theta,\phi) S_{k',n'}(r) Y_{l',m'}(\theta,\phi) d\mathbf{x} = \delta_{kk'} \delta_{nn'} \delta_{ll'} \delta_{mm'}$, which is required for the basis functions for $\mathbf{R}^3$. The regions where the basis functions have non-zero values are shown in Fig. 1. We refer to these basis functions as the “multi-layered spherical basis functions”, or MLSBFs, hereafter.

C. Series Expansion of a Scalar Field in Terms of the MLSBFs

Expansion of a scalar field $f(\mathbf{x})$ can be described as

$$f(\mathbf{x}) \approx \sum_{knlm}^{K,N,L} a_{k,n,l,m} B_{k,n,l,m}(r,\theta,\phi),$$
$$K \geq k \geq 0, N \geq n \geq 0, L \geq l \geq |m| \geq 0,$$

where the $a_{k,n,l,m}$ are the coefficients of the series, and *K*, *N*, and *L* are the maximum degrees or orders of the expansion. This representation of $f(\mathbf{x})$ means that we can express $f(\mathbf{x})$ with the $a_{k,n,l,m}$ by using these basis functions.

The coefficients $a_{k,n,l,m}$ can be calculated by taking the inner product of $f(\mathbf{x})$ and the corresponding basis function, i.e., $a_{k,n,l,m} = \int f(\mathbf{x}) B_{k,n,l,m}(r,\theta,\phi) d\mathbf{x}$. This equality is derived from the orthogonality of the basis functions. In our method, $a_{k,n,l,m}$ are computed by numerical integration.

D. Fast computation of an inner product

By using the expansion of a field, the inner product required for calculating a scoring function can be reduced to the inner product of the two coefficient vectors, which requires far less computation than numerically integrating the products of those two fields. That is,

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

Thus, the total scoring function, $E(\mathrm{T}^A, \mathrm{T}^B)$, can be reduced to

$$\begin{aligned}
E(\mathrm{T}^A, \mathrm{T}^B) &= \sum_{i=1}^{N_s} w_i \int f_i^{\mathrm{T}^A}(\mathbf{x}) g_i^{\mathrm{T}^B}(\mathbf{x}) d\mathbf{x} \\
&= \sum_{i=1}^{N_s} w_i \sum_{knlm} a_{i,k,n,l,m}^{\mathrm{T}^A} b_{i,k,n,l,m}^{\mathrm{T}^B}
\end{aligned}$$

where $a_{i,k,n,l,m}^{\mathrm{T}^A}$ and $b_{i,k,n,l,m}^{\mathrm{T}^B}$ are the coefficients of the transformed scalar fields $f_i^{\mathrm{T}^A}(\mathbf{x})$ and $g_i^{\mathrm{T}^B}(\mathbf{x})$ [1].

E. Fast Rotational and Translational Operations

As described above, we need to obtain transformed coefficients, $a_{k,n,l,m}^{\mathrm{T}^A}$ and $b_{k,n,l,m}^{\mathrm{T}^B}$, for a configuration space search. Those coefficients can be directly computed from the original coefficients, which is significantly faster than performing re-expansion of a field.

1) Direct rotational operation on coefficients

The coefficients of a rotated field $a_{k,n,l,m}^{\mathrm{R}}$ where R denotes the rotational operator on a field can be obtained from the original coefficients $a_{k,n,l,m}$ by using the equation $a_{k,n,l,m}^{\mathrm{R}} = \sum_{m'=-l}^{l} a_{k,n,l,m'} R_{m,m'}^{l}(\mathrm{R}^{-1})$, where $R_{m,m'}^{l}(\mathrm{R})$ denote the rotation matrices for real spherical harmonics analytically determined from R [19], [20].

2) Direct translational operation on coefficients

The coefficients of a translated field $a_{k,n,l,m}^{\mathrm{S}_{\Delta z}}$ where $\mathrm{S}_{\Delta z}$ denotes a translational operation along the *z*-axis with the shift of $(0,0,\Delta z)$ can be obtained by

$$\begin{aligned}
a_{k,n,l,m}^{\mathrm{S}_{\Delta z}} &= \sum_{k'n'l'} a_{k',n',l',m} \int_0^{\infty} \int_0^{\pi} S_{k',n'}(r') S_{k,n}(r) \\
& \quad \mathrm{P}_{l'}^{|m|}(\cos\theta') \mathrm{P}_{l}^{|m|}(\cos\theta) r^2 \sin\theta d\theta dr \\
&\equiv \sum_{k'n'l'} a_{k',n',l',m} O_{k',k,n',n,l',l,|m|}(\Delta z)
\end{aligned}$$

where $\mathrm{P}_l^m(x)$ denote the normalized Legendre polynomials and $O_{k',k,n',n,l',l,|m|}(\Delta z)$ denote the overlap integrals. The values of the $O_{k',k,n',n,l',l,|m|}(\Delta z)$ are calculated in advance using numerical integration, and the results are stored as a table for later use because they are independent of the scalar fields used for calculation. As one can see from the triple summations, this operation costs more than the rotational operation does.

By using the rotational operations and the translational operation along the *z*-axis, we can perform a six-dimensional search as described later.

F. Scoring Functions

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) [21]. Since a naive method to compute it in our framework requires 18^2 inner product calculations, which costs too much, we have reduced the number of terms to compute by using matrix diagonalization to the ACE energy matrix [18]. By applying the technique and selecting the two most contributing terms, we reduced the number of terms to use from 18^2 to 2.

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 configurations. The definition of the field is as follows:

[1] The suffix *i* is appended to the coefficients $a_{i,k,n,l,m}^{\mathrm{T}^A}$ and $b_{i,k,n,l,m}^{\mathrm{T}^B}$ to denote that they are the coefficients of the *i*-th term of a scoring function. This suffix is often omitted when the description is independent of the term number *i*.

$$\rho^X(\mathbf{x}) \equiv \begin{cases} v_{core} & \text{if } \mathbf{x} \text{ is inside a core atom} \\ v_{surface} & \text{if } \mathbf{x} \text{ is inside a surface atom} \\ & \text{(and not inside any core atoms)} \\ v_{vicinity} & \text{if } \mathbf{x} \text{ is in the vicinity of a surface atom} \\ 0 & \text{otherwise} \end{cases}$$

where v_{core} and $v_{surface}$ are positive weight values, and $v_{vicinity}$ is a negative weight value introduced for taking account shape complementarity in addition to atomic collision effect. An atom is classified as a surface atom if its solvent-accessible surface area is more than 1.0 Å^2. Otherwise, it is classified as a core atom. A coordinate $\mathbf{x}$ is defined to be in the vicinity of a surface atom *i* if the distance between $\mathbf{x}$ and the center of atom *i* is less than or equal to the van der Waals (vdW) radius plus the thickness of a skin, and if $\mathbf{x}$ is not inside any atoms. We are currently using 2.2 Å as the thickness of a skin.

By introducing $v_{vicinity}$, the term not only expresses shape complementarity but also creates a minute artifact of positive score that appears when the distance between atoms is around [sum of vdW radii + 2.2 Å ~ 4.4 Å]. However, the effect of this artifact is negligible for finding configurations with which molecules interact because it is much less than the effect of shape complementarity that arises when molecules are close enough to interact.

G. Outline of the Docking Procedure

1) Precalculations

The precalculation required before the docking computation is the creation of the look-up table for the overlap integrals, $O_{k',k,n',n,l',l,|m|}(\Delta z)$. This table is created for typical parameters of the maximum degrees/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 for which no tables have been created. As for each molecule, the process of obtaining its coefficients is required. This process is required only once per molecule.

2) Main docking process

The actual docking is performed by evaluating the scoring function of each configuration in the search space of five rotational dimensions and one translational dimension. This search space decomposition is equivalent to that of Ritchie et al. [16]. A brief explanation is shown in Fig. 2.

The basic procedure is described in Algorithm 1 in pseudo code. Although the search points are spread everywhere within a sphere of radius $z_{\max}$ in Algorithm 1, they can be limited in several ways to reduce the search space. Our current implementation includes the mechanism that excludes configurations which obviously do not fall into the near-native configurations such as ones with too much overlap, or ones in which two molecules are too far to interact.

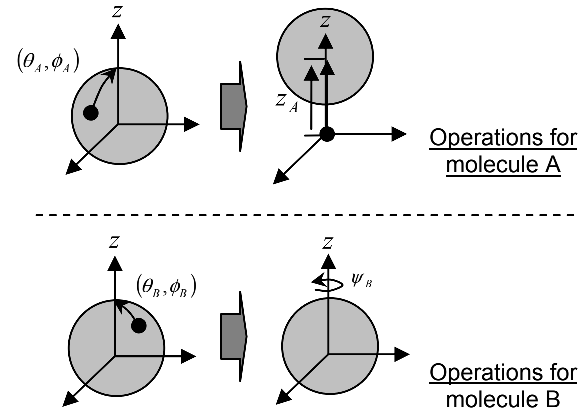

Figure 2. A configuration specified with $(\theta_A, \phi_A, z_A, \theta_B, \phi_B, \psi_B)$. Top: Rotational and translational operations for molecule *A*. The first rotation is an operation with which a unit vector determined by (θ_A, ϕ_A) is rotated to be parallel to the *z*-axis. The next operation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

Algorithm 1: Basic procedure of our docking process

```
$P := (p_1, \cdots, p_i, \cdots)$ where $p_i$ is one of the points evenly distributed on a unit sphere;
$R := (\mathrm{R}_1, \cdots, \mathrm{R}_i, \cdots)$ such that $\mathrm{R}_i$ is the rotational operation with which $p_i$ comes on to the z-axis;
$R^A_{\max}$ := maximum radius of the non-zero area of fields of a molecule A;
$R^B_{\max}$ := maximum radius of the non-zero area of fields of a molecule B;
$z_{\max} := R^A_{\max} + R^B_{\max}$ ;
foreach $\mathrm{R}^A$ in $R$ { ... (loop 1)
  for all i, set all $a'_{i,k,n,l,m}$ by applying operation $\mathrm{R}^A$ to $a_{i,k,n,l,m}$ ;
  for z:=0 to $z_{\max}$ with step $\Delta z$ { ... (loop 2)
    for all i, set all $a''_{i,k,n,l,m}$ by translating $a'_{i,k,n,l,m}$ along the z-axis by z;
    foreach $\mathrm{R}^B$ in $R$ { ... (loop 3)
      for all i, set all $b'_{i,k,n,l,m}$ by applying operation $\mathrm{R}^B$ to $b_{i,k,n,l,m}$ ;
      for $\psi_B := 0$ to $2\pi$ with step $\Delta\psi_B$ { ... (loop 4)
        for all i, set all $b''_{i,k,n,l,m}$ by rotating $b'_{i,k,n,l,m}$ around the z-axis by $\psi_B$ ;
          $E := \sum_{i=1}^{N_S} w_i \sum_{knlm} a''_{i,k,n,l,m} b''_{i,k,n,l,m}$ ;
          if E satisfies acceptance criteria for a possible candidate {
            output $(\mathrm{R}^A, z, \mathrm{R}^B, \psi_B, E)$;
          }
        }
      }
    }
}
```

To avoid the candidates being flooded with almost same configurations, we selected just three configurations from ones created within loop 4 in Algorithm 1 by using the only one best-scored configuration per 120 degrees of rotation (loop 4 performs rotational operations around the *z*-axis to molecule *B*).

TABLE I. PDB DATA USED IN COMPUTATIONAL EXPERIMENT

Complex	Mol. A	#residues	Mol. B	#residues
1UGH	1AKZ	223	1UGI(A)	84
1BRB	1BRA	223	6PTI	58
2SIC	1SUP	275	3SSI	113
2PTC	3PTN	223	6PTI	58
1CHO	5CHA(A)	241	2OVO	56
1CGI	1CHG	245	1HPT	56
2KAI	2PKA(AB)	232	6PTI	58

* The name of a molecule is represented with its PDB ID. Characters in parentheses denote chain IDs.

TABLE II. PARAMETERS USED IN EXPERIMENT

Thickness of a layer	a=12.5 Å		
Maximum deg./order used to get coefficients	Mol. A	k=0:	N=2,L=4
		k=1:	N=3,L=6
		k=2:	N=4,L=10
	Mol. B	k=0:	N=3,L=6
		k=1:	N=4,L=10
		k=2:	N=4,L=10
Granularity of search	$\Delta\theta_A, \Delta\phi_A, \Delta\theta_B, \Delta\phi_B$:		Approx. 15.8 deg. [a]
	$\Delta\psi_B$:		15.0 deg.
	Δz_A :		1.0 Å
Constants for steric hindrance	v_{core} =1.0, $v_{surface}$ =0.2, $v_{vicinity}$ =-0.1		
Weight for each term	1.0 for ACE, 750.0 for steric hindrance		

a. Derived from recursive subdivision of an icosahedron.

III. RESULTS AND DISCUSSION

A. Data and Parameters Used for Experiment

The structure data of proteins used for evaluation is listed in Table I. To perform unbound dockings, we used complex structures whose unbound structures of monomers were also available in the Protein Data Bank (PDB). The parameters used throughout this experiment are shown in Table II. We increased the numbers of coefficients used to express the layers for k=0,1 of the second molecules (molecule B) compared to those of the first ones (molecule A). It is because generally the surface area of molecule B is around the layers for k=0,1, in contrast with the case of molecule A of which the surface area is around the layers for k=1,2, which can be seen from the data of maximum k in Table III. This adjustment showed improvement in the quality compared to the case where we used the same number of coefficients on both molecules (data not shown), which became available thanks to the introduction of the MLSBFs.

B. Computation Time

We measured computation time required with a single 2.4 GHz Pentium 4 processor. The computation time for each case is shown in Table III. The total number of configurations generated and evaluated within the computation was around the order of 10^7. The difference in the computation time mainly came from the difference in maximum k for each molecule, which depends on the size of a molecule. Another minor factor is the difference in the shape of each molecule, as they cause difference in the number of configurations filtered out by the mechanism that excludes configurations which obviously do not fall into near-native configurations.

C. Qualities of the Predicted Structures

We measured the quality of the candidates of a complex structure by using the root-mean-square deviation (RMSD) of the interface residues, called the interface RMSD. This quantity is widely used for assessing the candidate structures of computational docking [22], [23], [14]. 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.

Table IV shows the best interface RMSD within a certain number of candidates, and Table V shows the rank of the first near-native structure for various thresholds. Table IV indicates that our program listed at least one near-native structure whose interface RMSD is less than 2.0 Å within the top 8000, and at least one structure whose interface RMSD is less than or close to 3.0 Å within the top 1000. This implies that if we are to apply a refinement method to the candidates generated with this method, using the top 8000 or 1000 candidates (depending on the quality that the refinement method requires) can be a good index.

TABLE III. COMPUTATION TIME REQUIRED FOR EACH PAIR OF MOLECULES

Mol. A	Mol. B	Max. k		Comp. time (min.)
		Mol. A	*Mol. B*	
1AKZ	1UGI(A)	2	1	1.59
1BRA	6PTI	2	2	2.69
1SUP	3SSI	2	2	2.89
3PTN	6PTI	2	2	2.70
5CHA(A)	2OVO	2	1	1.68
1CHG	1HPT	2	1	1.59
2PKA(AB)	6PTI	2	2	2.73

TABLE IV. BEST INTERFACE RMSD IN THE TOP RANKS

Complex	R_{best} [a] in 8000 (rank)	R_{best} in 1000 (rank)	R_{best} in 100 (rank)	R_{best} in 10 (rank)
1UGH	1.36 (173)	1.36 (173)	1.50 (16)	2.02 (1)
1BRB	1.04 (503)	1.04 (503)	1.86 (78)	4.46 (2)
2SIC	1.67 (58)	1.67 (58)	1.67 (58)	4.98 (2)
2PTC	0.81 (326)	0.81 (326)	2.30 (33)	2.78 (10)
1CHO	1.35 (4214)	2.73 (750)	6.23 (25)	6.35 (9)
1CGI	1.82 (4603)	3.14 (140)	3.67 (6)	3.67 (6)
2KAI	1.33 (1944)	1.41 (113)	2.04 (24)	6.15 (2)

a. R_{best}: best interface RMSD (Å).

TABLE V. RANK OF THE FIRST NEAR-NATIVE IN THE TOP RANK

Complex	Rank of the first near-native (interface RMSD (Å))			
	R_{thr} [a] ***=2.5 Å***	***R_{thr}=3.0 Å***	***R_{thr}=4.0 Å***	***R_{thr}=5.0 Å***
1UGH	1 (2.02)	1 (2.02)	1 (2.02)	1 (2.02)
1BRB	78 (1.86)	30 (2.57)	30 (2.57)	2 (4.46)
2SIC	58 (1.67)	58 (1.67)	14 (3.11)	2 (4.98)
2PTC	33 (2.30)	10 (2.78)	1 (3.06)	1 (3.06)
1CHO	1697 (2.24)	750 (2.73)	606 (3.85)	606 (3.85)
1CGI	4603 (1.82)	2651 (2.75)	6 (3.67)	2 (4.43)
2KAI	24 (2.04)	24 (2.04)	24 (2.04)	24 (2.04)

a. R_{thr} : threshold of interface RMSD used for deciding whether a structure is considered as a near-native one

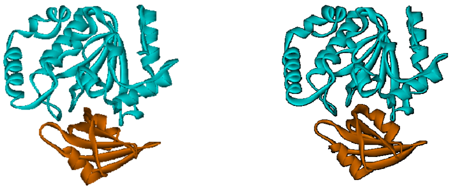

Figure 3. The predicted structure with the best interface RMSD within the top 1000 for 1UGH. The left shows the native structure and the right shows the predicted structure. (interface RMSD: 1.36 Å, rank: 173)

Fig. 3 shows one of our results for the unbound docking. It is the structure of a candidate with the best interface RMSD within the top 1000 for 1AKZ and 1UGI(A).

IV. Conclusion

We proposed a new protein-protein docking method using a series expansion of fields in terms of our novel multi-layered spherical basis functions for fast computations. Our computational framework has the same level of flexibility for designing the scoring function as the FFT-based methods. This work is basically an extension or generalization of our previous work [18]. The method proposed in [18] corresponds to the case where we only use a single layer of k=0, with a slight difference in the form of the basis functions. The basis functions we developed have improved the performance of approximating a field by enabling the efficient usage of coefficients, i.e., we can use more coefficients in important regions than in unimportant regions. Our experiments showed that our program was able to output predicted structures within a few minutes using a single 2.4 GHz Pentium 4 processor, and succeeded in listing at least one near-native structure whose interface RMSD is less than 2.0 Å within the top 8000 in all cases, and at least one structure whose interface RMSD is less than 2.5 Å within the top 100 in 5 out of 7 cases. We believe that we can further improve the performance by tuning the parameters and the scoring function, and we are also planning to include a refinement stage to the docking system.

Acknowledgment

This research is supported by grants-in-aid from the Institute for Bioinformatics Research and Development of the Japan Science and Technology Agency (BIRD-JST), and the Ministry of Education, Culture, Sports, Science and Technology (MEXT).

References

[1] I. Halperin, B. Ma, H. Wolfson, and R. Nussinov. Principles of docking: An overview of search algorithms and a guide to scoring functions. Proteins, 47(4):409–43, 2002.

[2] G. R. Smith and M. J. Sternberg. Prediction of protein-protein interactions by docking methods. Curr Opin Struct Biol, 12(1):28–35, 2002.

[3] D. W. Ritchie. Recent progress and future directions in protein-protein docking. Curr Protein Pept Sci, 9(1):1–15, 2008.

[4] I. D. Kuntz, J. M. Blaney, S. J. Oatley, R. Langridge, and T. E. Ferrin. A geometric approach to macromolecule-ligand interactions. J Mol Biol, 161(2):269–88, 1982.

[5] M. L. Connolly. Shape complementarity at the hemoglobin alpha 1 beta 1 subunit interface. Biopolymers, 25(7):1229–47, 1986.

[6] R. Norel, S. L. Lin, H. J. Wolfson, and R. Nussinov. Shape complementarity at protein-protein interfaces. Biopolymers, 34(7):933–40, 1994.

[7] R. Norel, S. L. Lin, H. J. Wolfson, and R. Nussinov. Molecular surface complementarity at protein-protein interfaces: the critical role played by surface normals at well placed, sparse, points in docking. J Mol Biol, 252(2):263–73, 1995.

[8] R. Norel, D. Petrey, H. J. Wolfson, and R. Nussinov. Examination of shape complementarity in docking of unbound proteins. Proteins, 36(3):307–17, 1999.

[9] J. Fernandez-Recio, M. Totrov, and R. Abagyan. Icm-disco docking by global energy optimization with fully flexible side-chains. Proteins, 52(1):113–7, 2003.

[10] M. Zacharias. Protein-protein docking with a reduced protein model accounting for side-chain flexibility. Protein Sci, 12(6):1271–82, 2003.

[11] J. S. Taylor and R. M. Burnett. Darwin: a program for docking flexible molecules. Proteins, 41(2):173–91, 2000.

[12] 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 U S A, 89(6):2195–9, 1992.

[13] H. A. Gabb, R. M. Jackson, and M. J. Sternberg. Modelling protein docking using shape complementarity, electrostatics and biochemical information. J Mol Biol, 272(1):106–20, 1997.

[14] R. Chen and Z. Weng. Docking unbound proteins using shape complementarity, desolvation, and electrostatics. Proteins, 47(3):281–94, 2002.

[15] B. S. Duncan and A. J. Olson. Applications of evolutionary programming for the prediction of protein-protein interactions. In Lawrence J. Fogel, Peter J. Angeline, and Thomas Baeck, editors, Evolutionary programming V : proceedings of the Fifth Annual Conference on Evolutionary Programming, pages 411–417. MIT Press, Cambridge, MA, 1996.

[16] D. W. Ritchie and G. J. Kemp. Protein docking using spherical polar fourier correlations. Proteins, 39(2):178–94, 2000.

[17] D. W. Ritchie, D. Kozakov, and S. Vajda. Accelerating and focusing protein-protein docking correlations using multi-dimensional rotational FFT generating functions. Bioinformatics, 24(17):1865–73, 2008.

[18] K. Sumikoshi, T. Terada, S. Nakamura, and K. Shimizu. A fast proteinprotein docking algorithm using series expansion in terms of spherical basis functions. Genome Inform, 16(2):161–73, 2005.

[19] C. H. Choi, J. Ivanic, M. S. Gordon, and K. Ruedenberg. Rapid and stable determination of rotation matrices between spherical harmonics by direct recursion. Journal of Chemical Physics, 111:8825–8831, 1999.

[20] J. Ivanic and K. Ruedenberg. Rotation matrices for real spherical harmonics. direct determination by recursion. J. Phys. Chem., 100(15):6342−6347, 1996.

[21] C. Zhang, G. Vasmatzis, J. L. Cornette, and C. DeLisi. Determination of atomic desolvation energies from the structures of crystallized proteins. J Mol Biol, 267(3):707–26, 1997.

[22] R. Mendez, R. Leplae, M. F. Lensink, and S. J. Wodak. Assessment of capri predictions in rounds 3-5 shows progress in docking procedures. Proteins, 60(2):150–69, 2005.

[23] R. Mendez, R. Leplae, L. De Maria, and S. J. Wodak. Assessment of blind predictions of protein-protein interactions: current status of docking methods. Proteins, 52(1):51–67, 2003.