

蛋白質-化合物ドッキングにおける グリッドの分散配置による配座探索の改良

伴 兼弘^{1,a)} 石田 貴士¹ 秋山 泰¹

概要: 蛋白質-化合物ドッキングソフトウェア Glide では、蛋白質表面を広く範囲にわたって探索する必要がある場合、標準では対象となる領域を包含する 1 つの大きなグリッドを配置するため探索に偏りが生じてしまう事がある。そこで本研究では、対象となる探索範囲を全て包含するように複数の小さなグリッドを分散させて配置する手法を提案し探索効率を向上させることに成功した。

キーワード: 蛋白質-化合物ドッキング, Glide, グリッド, 分散配置, 探索効率, 配座探索

Abstract: Glide protein-ligand docking algorithm often fails to find a correct binding mode because the search easily falls into local minima, when the search target area widely distributes on a protein surface and a large search grid is used. In this research, we propose a novel method to improve search efficiency in the case by arranging multiple small search grids.

Keywords: protein-ligand docking, Glide, grid, dispersedly setting, search efficiency, conformation search

1. はじめに

近年ドッキングシミュレーションは蛋白質と化合物の立体構造情報から、未知の結合部位を発見し結合様式まで予測することが試みられており、医薬品開発や生体メカニズムの解明に有用な道具となってきた。そのため、ドッキングシミュレーションの精度を向上させることはとても重要である。現在、蛋白質と化合物のドッキングシミュレーションの精度を向上させるためには大きく二つの問題がある。一つが結合自由エネルギーの評価、もう一つが配座探索の問題である。結合自由エネルギーの評価における問題は、厳密な物理学的な評価が計算量の問題から困難なため、疎視化したモデルを利用することに起因し、配座探索の問題は、探索すべき結合の状態（コンフォメーション）の数の多さに起因している。

特に配座探索においては、結合部位が未知の蛋白質と化合物の相互作用を解析する時に問題となる。つまり、蛋白質の結合部位が既知のものに関しては、その結合部位の周辺のみ配座探索をすればよい。しかし、既知の結合部位以外を狙いたいときや、結合部位が同定されていない蛋白質においては、蛋白質の表面全域に対して配座探索を

しなくてはならない。このため配座探索数が増えてしまい膨大な計算時間がかかってしまうという問題が生じる。そのため、一般的に結合部位が未知の蛋白質に対して、PocketFinder[1] や SiteMap[2] などのソフトウェアを用いて化合物の結合候補部位を予測する。そして蛋白質と化合物のドッキングシミュレーションでは、この予測された結合候補部位すべてをドッキングの探索範囲とする。現在、ドッキングシミュレーションを行う為のソフトウェアとしては、AutoDock[3], GOLD[4], FlexX[5], Glide[6] などが存在する。これらの中でも Glide は様々なベンチマークで良い精度を持つソフトウェアであることが示されている [7][8][9]。

Glide における通常のプロトコルでは、SiteMap により予測された結合候補部位を全て包含する一つの大きなグリッドを配置する。しかし、グリッドのサイズが大きくと探索範囲が広すぎると、Glide の探索アルゴリズムでは配座の探索に偏りが生じてしまうことがある。ここでいう探索の偏りとは、探索の初期段階で良いスコアが得られた配座の周辺を集中して探索することにより、その他のスコアにおける配座の周辺で最適解が見つかる可能性を排除してしまうことである。このとき、ドッキングの対象となる探索範囲を全て包含するように、複数の小さなグリッドを分散させて配置できれば、局所解に陥ることをできるだけ避けられ、より最適な配座を探索できると考えられる。

¹ 東京工業大学
Tokyo Institute of Technology

^{a)} ban@bi.cs.titech.ac.jp

そこで本研究では、グリッドを最適に分散配置させることで、効率的な探索を行う手法を提案する。その際、その最適化問題を集合被覆問題として扱い、実際にアルゴリズムを実装し配座探索を改良できることを確認する。

2. Glide の配座探索アルゴリズム

Glide の配座探索アルゴリズム [6] はリガンドがレセプターに結合するうえで、最も結合自由エネルギーが低くなる配座を見つけるためのものであり、計算手順は三つの段階に分けることができる。一つ目の段階では、入力されたリガンドの立体配座にもとづいて、化合物の結合候補点を判別する。二つ目の段階では、選定された結合候補点にリガンドの原子を配置し、スコアリングにより結合候補点を絞り込む。三つ目の段階では、結合自由エネルギーを最小化するように、剛体変換や回転を用いてリガンドのコンフォメーションを変化させ、エネルギーが低いものを選び出し、GlideScore[6] により最もスコアが良いものを選び出す。特に二つ目の段階におけるスコアリングによる篩い分けは、GreedyScore によるものと、Refinement によるものの二段階の篩い分けを行う。GreedyScore による篩い分けでは、結合候補点にリガンドの全原子を配置し、回転させながら ChemScore[10] と呼ばれるスコア関数によりスコアリングし、標準の設定では上位 5000 個の配座を選び出す。Refinement による篩い分けでは、GreedyScore による篩い分けで選定された結合候補点に対して、リガンドの中心を $\pm 1 \text{ \AA}$ ずつずらして ChemScore で再スコアリングし、標準の設定では上位 400 個の配座を選び出す。また、三つ目の段階におけるエネルギー最小化の過程では、剛体変換などを用いてエネルギーを最小化した後で、最もエネルギーの低い 3～6 個の配座を MonteCarloProcedure[6] によって結合のねじれ角などを最適化し GlideScore を用いてスコアを求め、最もスコアの良いものを選び出す。この一連の各プロセスにおける篩い分けによって選定される配座の数は、レセプターグリッドのサイズによらず一定である。

3. 提案手法

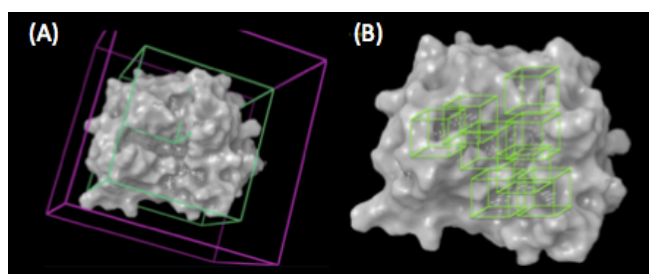


図 1 (A)Glide の標準の実装におけるグリッドの配置, (B) 提案手法のアルゴリズムを用いて実装した時のグリッドの配置

Fig. 1 (A)The placement of a grid in the Glide standard, (B)The placement of grids with our method

Glide の標準ではドッキングの対象となる領域を包含する 1 つの大きなグリッドを配置するため探索に偏りが生じてしまうことがある。この問題点に対して本研究では、ドッキングする対象となる探索範囲にグリッドを分散させて配置する手法を提案する。その際、ドッキングの対象となる探索範囲に対してグリッドを分散させて配置する問題を集合被覆問題に帰着させる。ここで、集合被覆問題とは n 個の要素からなる台集合 U 、 U の部分集合の族 $S = \{S_1, S_2, \dots, S_l\}$ およびコスト関数 $c: S \rightarrow \mathbb{Q}_+$ が入力として与えられたとき、 U の要素をすべてカバーするコスト最小の S の部分族を求める問題である [12]。今回の問題においては、ドッキングの対象となる探索範囲を SiteMap により得られたすべての Site-point とし、これを集合被覆問題における台集合とする。ここで Site-point とは、レセプター上でリガンドが結合するのに適切な点のことである。また台集合の各要素を中心とする大きさ一定のグリッドの集合を、集合被覆問題における部分集合の族とする。ここでコスト関数は各グリッドの INNERBOX (1 における緑色のボックス: Glide の探索アルゴリズムにおける探索範囲であり、リガンドの中心が含まれる) に含まれる台集合の要素数とする。このようにドッキングの対象となる探索範囲に対してグリッドを分散させて配置する問題を集合被覆問題に帰着させた。このとき、集合被覆問題は NP 困難な問題として知られている [11] ため近似アルゴリズムとしてグリーディーアルゴリズム [12] を応用する。アルゴリズムの概要は、(1)SiteMap により得られた Site-point の集合 S を入力とする。(2) 空集合 C を用意する。(3) S の要素からコストが最も高いグリッド G を一つ選ぶ。(4) G の中心点を集合 C に加え、一方で (5) S から G に含まれる要素をすべて削除する。そして (6) S が空集合になるまで (3)～(5) を繰り返す。(7) 集合 C を分散配置に用いるグリッドの中心点とする。詳しいアルゴリズムの内容を 2 に示した。また計算量については、SiteMap により得られた Site-point の数を n とするとき、 $O(n^3)$ 程度である。提案手法のアルゴリズムは、Glide cross docking (以下、「XGlide」) [13] のスクリプト xglide.py を改変することにより実装されている。

4. 評価実験

今回の評価実験では、探索範囲を包含する一つのグリッドを配置する Glide の標準よりも、本研究で提案するドッキングする対象となる探索範囲にグリッドを分散させて配置する手法の方が探索効率が良いことを確認する。探索の主要な評価は、DockingScore と計算時間を用いた。

4.1 データセットの取得と準備

CCDC/Astex data set と呼ばれる蛋白質とリガンドの複合体で構成されたデータセット [14] を用いた。本研

アルゴリズム

```

1: SiteMap で得られた Site-point の集合を S とする;
2: While (S が空集合になるまで繰り返す) do
3:   for (S の要素を 1 つ選択し gc とする) do
4:     for (S の要素 s を全て選択するまで繰り返す) do
5:       if (gc を中心とする INNERBOX に s が含まれる) then
6:         集合 A に s を追加する;
7:       end if
8:     end for
9:   if (集合 A の要素数が今まで調べたもののうち最大) then
10:     現在の gc をグリッド中心の候補とする;
11:     現在の集合 A の要素を集合 B に上書きする;
12:   end if
13:   集合 A を空集合にする;
14: end for
15: 集合 C に gc を追加する;
16: S から集合 B の要素をすべて削除する;
17: end while
18: 集合 C の各要素を中心とするグリッドを作成する;

```

図 2 提案手法のアルゴリズム

Fig. 2 The algorithm of proposed method

究では、このデータセットの中から計算量の問題のため 20 個の複合体を無作為に選び、それぞれの立体構造情報を Protein Data Bank (以下、「PDB」) [15] よりダウンロードした。本研究で用いた PDB ID はそれぞれ 1A4G, 1AJ7, 1B9V, 1DBB, 1EJN, 1FAX, 1FKG, 1HDC, 1IBG, 1MMQ, 1QBR, 1RNE, 1TPH, 1XKB, 2DBL, 2H4N, 2TMN, 2YPI, 3ERD, 7CPA である。ただし、初めに無作為に選んだものに 1GPY, 1RT2, 4CTS があったが、ドッキングの対象となる探索範囲を包含する一つのグリッドを作成する際に、探索範囲が広すぎるためにエラーとなってしまうため除いた。その代わりとして、新たに無作為に選んだ 1EJN, 1FAX, 2TMN をデータに加えた。次ぎに各複合体は Schrodinger 社のソフトウェア Maestro 上でリガンドとレセプターに分離した。分離後にレセプターに関しては、Glide の標準機能である Protein Preparation Wizard を用いて、水分子や金属原子などの補因子をすべて取り除き、蛋白質の立体構造を Preprocess, Optimize, Remove waters, Minimize の順序で最適化した。リガンドの配座発生, SiteMap, レセプターグリッドの作成および Ligand Docking は XGlide を用いて機械的に実行した。

4.2 SiteMap による探索範囲の絞り込み

ドッキングの対象となる探索範囲は Glide の標準機能である SiteMap を用いて決定する。SiteMap は、入力されたレセプターの立体構造に対して、結合部位の特徴を新規かつ迅速に発見するというものである。このとき SiteMap の設定において、標準でのサイトの報告数は最大で 5 個となっているが、出来るだけ多くの領域を探索するため、今回の実験ではいずれも最大で 10 個に変更した。また、その他のオプションの設定は初期設定のままとした。本研究では、SiteMap より得られたすべての Site-point をドッキングの対象となる探索範囲と定義した。

4.3 グリッドの作成について

蛋白質の立体構造と SiteMap により得られた Site-point の集合を用いることにより、探索範囲の全域に対して、エネルギー的性質を含めた情報を持つレセプターグリッドを作成する。この過程では Glide の "Glide Grid Generation" 機能を用いた。今回の実験では Glide Grid Generation の設定は、標準の実装と提案手法による実装とで一部異なる。設定の違いは、SiteMap で得られたサイトに対するグリッドの中心の取り方、グリッドの OUTERBOX (1 における紫色のボックス: 配座探索においてリガンドの原子がはみ出しはけない範囲) および INNERBOX のサイズの 2 点である。他のオプションに関しては、初期設定のままとした。グリッドの中心の取り方に関しては、「標準の実装」ではサイトの重心をグリッドの中心とした。グリッドのサイズに関しては、INNERBOX の辺の長さはサイトを包含する最小の自然数とし、OUTERBOX の辺の長さは INNERBOX の辺に一定値 (今回の実験では 16 Å) を加えた値とした。ただしグリッドの形状はいずれの場合も立方体であるため、三辺とも辺の長さは同じである。ここで、辺の単位は原則 Å を用いている。一方で、提案手法のアルゴリズムを用いた実装においては、アルゴリズムにより得られたグリッドの中心点に、大きさが一定のものを複数の作成する。グリッドの INNERBOX と OUTERBOX の辺の長さは、それぞれ 10 Å と 26 Å と一定にした。これら三つの違いにより、標準の実装では SiteMap により得られたサイトの個数だけレセプターグリッドが生成され、提案手法を用いた実装ではサイトの個数以上のレセプターグリッドが生成される。ただし、ドッキングの対象となる探索範囲をすべて含むということに関してはどちらも満たしている。

4.4 Glide による蛋白質-化合物ドッキング

Glide における蛋白質-化合物ドッキングは、リガンドの初期配座により結果が異なるため、ドッキング計算を行う前に、分離して得られたリガンドに対して複数の配座を LigPrep によって発生させた。今回の実験において LigPrep のオプション設定は初期設定のままとした。この処理で得られた複数のリガンドの配座とレセプターグリッドを用いて、Ligand Docking によって蛋白質-化合物ドッキングを行う。このとき、一つのレセプターグリッドに対して一つのリガンドの配座が選択され、2.1 で述べた探索アルゴリズムにより、結合自由エネルギーが最適になると予想される配座を見つける。また予測モード以外の設定は初期設定のままとした。Glide のドッキング予測のモードには、詳細だが低速な XP モードと XP モードほど詳細ではないが高速な SP モードがある。今回の実験では、計算量の問題から SP モードのみを用いる。

本研究では、グリッドを分散させて配置することで探索

効率の向上を目指したが、より良い探索結果を得るためには配座探索数に関したパラメータを変更し、より多くの探索を行うことも可能である。したがって、標準のグリッド設定では、探索を詳細にして提案手法の計算時間よりも長い時間を探索に費やしても、より良いスコアの配座を見つけられないことを示すため、今回の実験では配座探索数に関する二つのパラメータの変更と、より厳密に探索するためのオプションである Expanded Sampling[16]を設定したものに対しても「拡張探索」として比較を行った。パラメータの設定に関しては、標準の実装において GreedyScore による篩い分けで選定される標準 5000 個と、Refinement による篩い分けで選定される標準 400 個の値を計算時間が提案手法の計算時間より長くなるまで、各パラメータをそれぞれ 10 倍ずつ増やしている。すなわち、1A4G, 1EJN, 1FAX, 1IBG, 1MMQ, 1QBR, 1RNE, 1XKB, 2DBL, 2H4N, 2TMN, 7CPA においては 10 倍、1AJ7, 1B9V, 1HDC, 1TPH, 2YPI, 3ERD においては 100 倍、1DBB, 1FKG においては 1000 倍として設定した。また標準の実装において、配座探索数に関するパラメータの設定を初期設定から変更しない場合も実験をした。

4.5 実験結果

ここでは Glide において、配座探索数に関するパラメータの設定を初期設定から変更しない標準、提案手法および Expanded Sampling のオプションを加えパラメータの変更した拡張探索における DockingScore, RMSD, 計算時間の結果をそれぞれ 3 に示す。Docking Score は Glide Score とほぼ同一の値であり、違いは Epik state penalties[18] がスコアに含まれるか否かであり、一般にドッキングの後処理でよく用いられる [17]。そのため本研究においてもこの Docking Score で評価している。Docking Score に関しては、ドッキングの対象としている探索範囲の全域において、結合自由エネルギーの値が最も低いものを示してある。RMSD は 2 つの分子における対応する原子間の平均二乗誤差であり、一般にドッキングシミュレーションの結果は RMSD によって評価されるが、RMSD を探索手法の評価に用いてしまうと、より良いスコアの配座を探索できたとしても RMSD がよくないという可能性がある。これは、スコア関数の問題であるため、今回の実験において、探索手法は DockingScore を主要な評価対象とした。3 の結果から、DockingScore の平均値をみた場合、標準では -7.95、提案手法では -8.73、拡張探索では -8.60 となった。このことは、提案手法が上述の三つの探索手法の中で最も平均の DockingScore が良く（結合自由エネルギーが低く）なった。また各複合体に対して、提案手法の DockingScore が標準の DockingScore よりも低くなっている割合は 18/20 であり、提案手法の DockingScore が拡張探索の DockingScore よりも良くなっている割合は 14/20 であった。計算時間に

ついては、ジョブが開始されてからすべてのプロセスが終了するまでの時間を示してある。この計算時間には、集合被覆問題をグリーディーアルゴリズムで解く時間も含まれているが、この時間は数秒程度であり全体に対して無視できるため別々で計算せずに、そのままの値で比較する。平均計算時間に関しては、標準が 3587.3 秒、提案手法が 9053.3 秒であり、拡張探索が 13996.7 秒となった。これより、提案手法は標準の探索より 3 倍程度の計算時間が必要となっている。一方で、拡張探索は提案手法よりも 1.5 倍以上の計算時間が必要である。

4.5.1 RMSD の比較

次に RMSD については、Schrodinger 社のソフトウェア Maestro の標準機能 Superposition を用いて計算を行った。また計算の対象は、結晶構造の配座と、ドッキング計算により得られた DockingScore が最も低いリガンドの配座の二つである。ここでは、標準、提案手法および拡張探索における、RMSD を用いた予測精度の比較を行った結果を図 3 に示した。図 3 において、黒色の線が標準、橙色の線が提案手法、青色の線が拡張探索を用いたときの結果である。図 3 の縦軸は正解率を表し、横軸は RMSD の上限値を示している。ここで、正解率とは結晶構造のリガンドと、ドッキング計算により得られた DockingScore が最も高い複合体のリガンドの RMSD が、指定された RMSD よりも小さいものの全体に対する割合と定義している。また、RMSD の上限値とは、指定された RMSD のことである。RMSD の上限値は、0Å から 10Å まで 0.5Å 刻みで示している。図 3 の結果からは、RMSD 上限値 2Å において、標準の場合は全体の 40% が上限値内に含まれていた。また提案手法を用いた場合、全体の 55% が含まれており標準の探索より提案手法の方が良い予測結果を返している。一方、拡張探索も同様に標準の探索より良い予測結果を返しており、特に RMSD の上限値が 3.5Å 以上において、拡張探索の正解率の方が、提案手法の正解率を上回る結果となった。

5. 考察

5.1 探索効率の向上について

DockingScore の平均値をみた場合、標準では -7.95、提案手法では -8.73、拡張探索では -8.60 と三つの探索手法の中で提案手法の値が最も良い値となった。また各複合体に対して、提案手法の DockingScore が標準および拡張探索の DockingScore よりも良くなっている割合は、それぞれ 18/20 と 14/20 であることから、提案手法が標準および拡張探索よりも探索手法として優位であることが示唆される結果となったが、その差が統計的に優位であるかを確認するため、提案手法と標準および提案手法と拡張探索におけるスコアの組をそれぞれ 2 標本として、ウィルコソンの符号付順位和検定 [19] を行った。ここで、ウィルコソンの符号付順位和検定とは、ノンパラメトリックな分布を仮

PDB	標準			提案手法			拡張探索		
	Score[kcal/mol]	RMSD[Å]	time[s]	Score[kcal/mol]	RMSD[Å]	time[s]	Score[kcal/mol]	RMSD[Å]	time[s]
1A4G	-7.34	23.9	6072	-8.09	23.1	14012	-7.90	22.8	47629
1AJ7	-7.33	1.8	1089	-7.74	2.3	4933	-7.79	1.7	8110
1B9V	-6.16	22.9	759	-8.97	23.4	2019	-7.06	23.5	5737
1DBB	-8.74	0.5	1406	-9.26	0.6	4476	-8.72	0.5	1835
1EJN	-6.72	12.3	577	-8.76	0.9	1264	-7.56	1.2	1507
1FAX	-9.21	4.7	904	-9.46	8.9	1807	-8.64	4.5	5067
1FKG	-7.77	1.9	137	-7.90	1.6	6390	-8.05	1.9	1656
1HDC	-8.07	6.1	1046	-7.97	6.1	3661	-8.09	6.1	4935
1IBG	-8.66	2.3	6937	-8.88	1.1	23745	-8.64	2.3	26969
1MMQ	-7.88	9.5	207	-8.18	0.4	550	-7.45	0.4	1025
1QBR	-7.91	10.6	1176	-10.18	7.8	2131	-14.24	1.2	4162
1RNE	-13.86	1.5	44045	-14.48	1.2	97028	-15.19	1.0	133134
1TPH	-6.01	1.1	457	-6.57	0.9	1092	-6.01	1.1	4243
1XKB	-7.87	8.9	1006	-11.54	2.1	2745	-11.44	2.1	3366
2DBL	-8.67	1.1	2421	-9.25	1.3	7235	-8.69	1.1	11226
2H4N	-5.02	6.3	629	-5.45	6.3	1352	-5.07	6.2	2592
2TMN	-5.16	2.6	577	-5.33	1.3	1264	-5.08	3.5	2180
2YPI	-8.16	0.8	590	-7.95	1.1	1829	-8.07	0.8	3150
3ERD	-9.87	0.5	622	-9.91	0.5	1437	-9.87	0.5	3707
7CPA	-8.75	4.6	1089	-8.80	4.4	2096	-8.55	4.0	7705
Average	-7.96	6.2	3587.3	-8.73	4.8	9053.3	-8.61	4.3	13996.75

図 3 標準, 提案手法, 拡張探索における Score(DockingScore), RMSD, 計算時間の結果

Fig. 3 Comparison of the results of three methods.

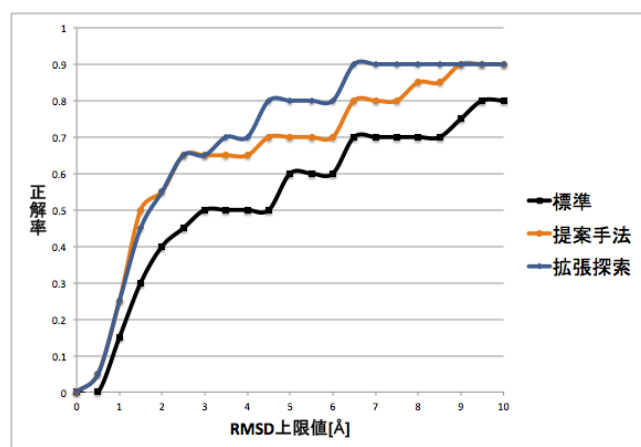


図 4 標準, 提案手法, 拡張探索における DockingScore が最も良かった配座と結晶構造の配座における RMSD が RMSD 上限値に含まれる割合 (正解率)

Fig. 4 The rate curve of RMSD for three methods.

定し, 対応のある 2 標本の組について, 母代表値に差があるか判定する検定手法である. R 3.0.0 を用いて検定を行った結果, 提案手法と標準についての P 値は 6.2×10^{-5} となり, 提案手法と拡張探索についての P 値は 0.049 となった. このことから, 有意水準 0.05 において, 提案手法と標準および提案手法と拡張探索のスコアの 2 標本の母代表値にはどちらも差があると判定された.

また拡張探索の平均計算時間は, 提案手法の平均計算時間の 1.5 倍以上であることから, より高いスコアを探索できるという意味において, 提案手法は Glide で実装されている標準の配座探索手法よりも探索効率が優れていると言

える. 従って, ドッキングの対象となる探索範囲を全て包含するように複数の小さなグリッドを分散させて配置する手法により, 配座探索を標準の実装に比べ改良できることが確認された.

今回の実験で, 最も提案手法の探索が成功した例は, PDB ID が 2TMN の結果である. 計算時間に関しては, 提案手法では 1264 秒に対し, 拡張探索では 2180 秒と 2 倍近くかかっている. その一方で, Docking Score は提案手法では -5.33 と拡張探索の -5.07 よりよいスコアが得られた. また RMSD においても, 提案手法が 1.3Å であるのに対し提案手法が 3.4Å と, 提案手法の方がより結晶構造に近い配座が得られた.

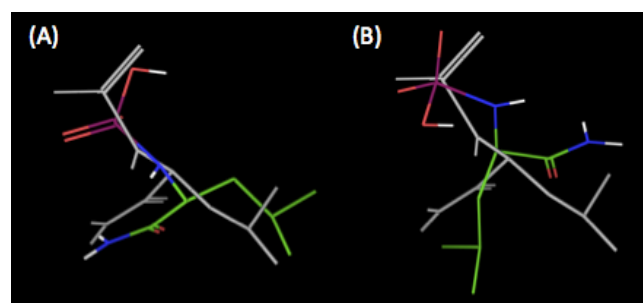


図 5 PDB ID: 2TMN に対するドッキングシミュレーションの結果: (A) 提案手法により得られた配座, (B) 拡張探索により得られた配座, 灰色の配座は結晶構造により得られた配座を示す

Fig. 5 The result of 2TMN : (A)The conformation that is obtained by using our method, (B)The conformation that is obtained by using the expanded explore. And the gray conformation is getting by X-ray.

5.2 提案手法の探索効率が良くなる要因について

提案手法と拡張探索の方法による相違点は、レセプターグリッドを小さく複数個に分けて作るか、一つの大きなものを作るかである。ここで、Glide の蛋白質-化合物ドッキングにおいて、一つのグリッドに対して、一つの化合物の配座探索数は一定である。そのためパラメータの設定により、拡張探索の配座探索数は、GreedyScore と Refinement における篩い分けの過程において、提案手法よりも多くなるように各複合体ごとに設定してある。この相違点により、提案手法が拡張探索よりも有利になる点は、プロセスの全体をみたとき GridMinimization[6] および MonteCarloProcedure に回される配座の数が多くなることである。これに関しては、単純に最適化される数が多ければ多いほど、より高いスコアを探索できる可能性は高くなると考えられる。また探索範囲を局所的に限定することで、全探索範囲の中で大域的最小点を集中して探索してしまうことを避け、最適解を探索することも有利な点である。これに関しては、大域的最小点の付近にある最適解はもちろん探索できるが、その範囲外に最適解が合った場合に、提案手法の方が標準の実装よりも多く探索が行えるため、最適解が見つかりやすいと考えられる。探索効率に関しては、GreedyScore や Refinement により選ばれる配座数は、最適解が含まれる範囲内で少ない方が探索効率は良くなる。この点において、標準の実装ではより良い解を探索しようとした場合、現状では GreedyScore や Refinement の配座探索数を増やす事でしか対応できないため探索効率は落ちてしまう。しかし、提案手法では、GreedyScore や Refinement の配座探索数を増やさず、ドッキングの対象となる探索範囲を探索できるため、探索効率が標準の実装よりも良くなったと考えられる。

5.3 RMSD による評価の結果について

RMSD を比較した結果において、4 に示すように提案手法よりも拡張探索の方がより結晶構造に近い配座を予測できたことを示している。このような結果となった要因は、DockingScore に関係していると考えられる。つまり、DockingScore と RMSD の相関が十分でなかった為に、より良いスコアを探索できる提案手法が標準の実装よりも予測精度が低くなったと考えられる。今後の研究では、検証データの数を増やして実験してみるということと、DockingScore と RMSD の相関を十分強くすることを目的とし、DockingScore を補正する事が次の重要な課題である。

6. 結論

本研究では、Glide を用いたドッキングシミュレーションにおいて、グリッドのサイズが大きく探索範囲が広すぎると、Glide の探索アルゴリズムでは配座の探索に偏りが

生じてしまう問題について、小さなグリッドを分散させて配置できれば、局所解に陥ることをできるだけ避けられ、より最適な解を探索することができるのではないかと考えた。そこでグリッドを分散させて配置させるアルゴリズムを実装し、標準の実装よりも探索効率が良くなることを確認した。今回の実験での探索効率の評価は、より短い探索時間でより良い DockingScore が得られるかどうかで判断している。そのため、標準の実装に関しては、提案手法よりも計算時間が長くなるように配座探索数に関係するパラメータを設定し、提案手法を用いた方がそれでもよい DockingScore を得られるのかを検証した。その結果、ドッキングの対象となる探索範囲を全て包含するように複数の小さなグリッドを分散させて配置する方法は、標準および配座探索数を増やした拡張探索の実装に比べ探索効率が良くなる事が確認できた。一方で、RMSD の比較においては拡張探索の実装の方が良くなった。これは DockingScore と RMSD の相関が十分でなかった為であると考えられ、今後は DockingScore の補正について考え、ドッキングの精度向上に向けた研究も同時に進める事が必要と考えられる。

謝辞 本研究は、文部科学省 博士課程教育リーディングプログラム東京工業大学「情報生命博士教育院」の支援を受けて行われたものである。

参考文献

- [1] Laurie, Alasdair TR, and Richard M. Jackson. "Q-SiteFinder: an energy-based method for the prediction of protein-ligand binding sites." *Bioinformatics* 21.9 (2005): 1908-1916.
- [2] Halgren, T. A. : "New Method for Fast and Accurate Binding-site Identification and Analysis", *Chem. Biol. Drug Des.*, 2007, 69, 146.
- [3] Goodsell, David S., Garrett M. Morris, and Arthur J. Olson. "Automated docking of flexible ligands: applications of AutoDock." *Journal of Molecular Recognition* 9.1 (1996): 1-5.
- [4] Jones, Gareth, et al. "Development and validation of a genetic algorithm for flexible docking." *Journal of molecular biology* 267.3 (1997): 727-748.
- [5] Rarey, Matthias, et al. "A fast flexible docking method using an incremental construction algorithm." *Journal of molecular biology* 261.3 (1996): 470-489.
- [6] Friesner, Richard A., et al. "Glide: a new approach for rapid, accurate docking and scoring. 1. Method and assessment of docking accuracy." *Journal of medicinal chemistry* 47.7 (2004): 1739-1749.
- [7] von Korff, Modest, Joel Freyss, and Thomas Sander. "Comparison of ligand- and structure-based virtual screening on the DUD data set." *Journal of chemical information and modeling* 49.2 (2009): 209-231.
- [8] Kellenberger, Esther, et al. "Comparative evaluation of eight docking tools for docking and virtual screening accuracy." *Proteins: Structure, Function, and Bioinformatics* 57.2 (2004): 225-242.

- [9] Marcou, Gilles, and Didier Rognan. "Optimizing fragment and scaffold docking by use of molecular interaction fingerprints." *Journal of chemical information and modeling* 47.1 (2007): 195-207.
- [10] Eldridge, M. D.; Murray, C. W.; Auton, T. R.; Paolini, G. V.; Mee, R. P. Empirical scoring functions: I. The development of a fast empirical scoring function to estimate the binding affinity of ligands in receptor complexes. *J. Comput.-Aided Mol. Des.* 1997, 11, 425-445.
- [11] M.R.Garey and D.S. Johnson, *"Computers and Intractability: A Guide to the Theory of NP - Completeness"*, W.H. Freeman and Company (1979)
- [12] V.V. ヴァジラーニ 著, 浅野 孝夫 訳: "近似アルゴリズム" シュプリンガー・フェアラーク東京株式会社 p15-p26
- [13] <http://www.schrodinger.com/scriptcenter/#Docking>
- [14] Nissink, J. Willem M., et al. "A new test set for validating predictions of protein?ligand interaction." *Proteins: Structure, Function, and Bioinformatics* 49.4 (2002): 457-471.
- [15] <http://www.rcsb.org/pdb/home/home.do>
- [16] [http : //helixweb.nih.gov/schrodinger](http://helixweb.nih.gov/schrodinger) - 2013.3 - docs/glide/glide_user_manual.pdf, p58
- [17] [http : //www.schrodinger.com/kb/348](http://www.schrodinger.com/kb/348)
- [18] <http://www.schrodinger.com/newsletter/12/68/>
- [19] Wilcoxon, Frank. "Individual comparisons by ranking methods." *Biometrics bulletin* 1.6 (1945): 80-83.