

タンパク質立体構造研究支援のための並列統合解析システムの構築

鬼塚 健太郎[†] 野口 保[†]
斎藤 稔[†] 秋山 泰[†]

並列応用プログラミングの一つとして、分子生物学の分野で重要なタンパク質立体構造研究を支援する並列統合解析システムを構築した。このシステム“PAPIA (PARallel Protein Information Analysis) System”は、タンパク質の立体構造のデータベースである Protein Data Bank (PDB) を管理するデータベース管理モジュールと、構造解析に必要な幾何学計算、統計処理、行列演算、3次元グラフィック処理を行なうライブラリを、並列処理のためのノード間データ転送機能とともに統合したものである。

本システムは、この分野の研究を行なう上で必要な解析のプログラミングを迅速に行ない、かつ並列処理によって高速に解析を行なって、結果を3次元グラフィックスで明解に表示することを可能とする。

An Integrated Parallel System for Protein Structure Analysis

KENTARO ONIZUKA,[†] TAMOTSU NOGUCHI,[†] MINORU SAITO[†]
and YUTAKA AKIYAMA[†]

We developed, as a parallel application program, an integrated parallel system for protein structure analysis, which is one of the most important fields in molecular biology. This system, “PAPIA (PARallel Protein Information Analysis) System,” consists of the database management module for PDB (Brookhaven Protein Data bank, *i.e.* a collection of protein-structure data), Three-dimensional geometric calculation, statistical analysis, matrix, and 3D graphics library, together with parallel programming facilities for the internode data-transfer.

The researchers in the field can easily develop a relatively effective parallel program for their particular structure analysis purposes, and can obtain and 3D-visualize the result of analysis quickly using the libraries provided by PAPIA system.

1. システム開発の背景と目的

1.1 タンパク質立体構造解析

タンパク質は、生物の生命現象の中で分子的視野で見たときに、もっとも重要な役割を担っている物質である。細胞内染色体のゲノム (DNA) 中に存在する遺伝子に DNA のコドン配列としてコードされているタンパク質のアミノ酸残基配列は、そのタンパク質が必要とされる条件になったときに読み出され、そのコドン配列から実際のアミノ酸が順に合成される (合成されたアミノ酸はアミノ酸残基とよばれる)。結合したアミノ酸残基の鎖 (ポリペプチド鎖) は、折れ畳まってタンパク質として機能できるようになる。タンパク質はアミノ酸残基配列が異なれば異なる立体構造を形成し、かつ異なる機能をもつ。すなわち、タンパク質の機能、特性を決定づけているのが、立体構造である。

進化論的な意味で、最適化され収束した結果として存在するタンパク質のアミノ酸残基配列と、それが折れ畳

まって形成された立体構造との関係は、単純な分子の化学結合式とその立体構造の関係のような物理化学的な因果関係ではなく、生命の進化の中で取捨選択された複雑な関係なのである。このタンパク質のアミノ酸配列と立体構造との関係、及び立体構造形成の仕組みの解明は、分子生物学の中でもっとも重要な研究分野である。

この分野に、強力な並列処理技術を応用した統合研究支援環境を導入することは、タンパク質立体構造予測問題の解決を早め、分子生物学の進歩発展、ひいては生命現象そのものの解明につながり、医療、製薬、品種改良、新素材開発などの分野にも大きな波及効果を及ぼすことにつながる。

1.2 従来のタンパク質立体構造解析システム

タンパク質の立体構造を解析するソフトウェアパッケージは、商用のものも含めて、いくつか存在する (たとえば InsightII¹⁾ など)。しかし、一般的にこれらのシステムは、一つのタンパク質について詳細な研究を行なっている生物学者にとって便利にできている。したがって特定のタンパク質の解析を行ないたい研究者が、特定のタンパク質を詳細に解析する用途においては大変

[†] 新情報処理開発機構

Real World Computing Partnership

有効である。しかし、これらのシステムで、構造既知のタンパク質全体について統計処理などを行なうことは、基本的に不可能であると言える。

今回の開発目的は、一つ一つのタンパク質に関して詳細な解析を行なうよりは、構造既知のタンパク質全てについて同一の解析を行ない、結果を統計的に処理するためのシステムの構築である。

1.3 タンパク質立体構造解析のための要素技術

1.3.1 PDBの読み取り技術

タンパク質の立体構造解析を行なう上で、現状でもっとも面倒な問題は、提供されている唯一のタンパク質立体構造データベースである Protein Data Bank(PDB)²⁾が、プログラムで自動解析するのが困難なデータセットであるということである。実験的なタンパク質立体構造決定法が確立した直後からのデータが玉石混淆の状態に網羅されている。したがって、データの質、内容、さらには記述方式も含めてかなり雑多なものである。タンパク質立体構造解析支援システムを作る上で、最初にやるべきことは、PDBを正しく読み取ることができる読み取り技術の確立である(このような目的のための研究開発の例としては、たとえば文献³⁾⁴⁾⁵⁾などがある)。

1.3.2 三次元幾何学解析

タンパク質の立体構造を解析するときには、その構造内の特定の原子間の距離や、構造内部のいくつかの原子によって定義される、結合角、二面角などの解析が必要である。また、三次元回転や並進移動や、類似した構造の三次元的な最適重ね合わせて(Least RMSD Fitting i.e. Least Root-Mean-Square-Deviation Fitting)、構造間の類似度計測も必要になる。これらの処理は高度なアルゴリズムを用いたプログラミングを必要とする。これら立体構造解析の要素となるプログラムをライブラリとして提供すれば、上述のPDBの読み取り部分と統合させて、有用な解析環境を構築できる。

1.3.3 統計解析

PDBに含まれているタンパク質の立体構造の数は、1997年9月現在、エントリー数で約6000である。タンパク質の中の特定の種類のアミノ酸残基がどのような構造を取っているかなどを解析するためには、統計的な処理が不可欠である。その中でも、二つ以上の量の相関や主成分分析などの解析をする機能が重要である。

1.3.4 三次元グラフィック表示

タンパク質の立体構造解析をする上では、研究者が具体的に三次元の物体として対象を操作することできるように、三次元グラフィックスによる操作環境が必要である。この機能は、統計処理で得られた結果をわかりやすく表示するときにも有効である。

1.3.5 配列解析

タンパク質の立体構造とアミノ酸残基配列との関係を

考えるときに、配列解析の機能も必要である。特に重要なのは、異なる残基配列をもつタンパク質間の配列上の相同性(類似度)と、立体構造上の相同性の相関解析において必要なアミノ酸残基配列のアライメントの機能である。

1.3.6 並列処理

タンパク質の立体構造解析は、これまで述べてきたタンパク質立体構造データベースの読み取り、三次元立体解析のアルゴリズム、統計処理、配列解析などを統合して行なうものである。特に、PDBの全エントリーに対して同一な処理を行なう場合が多い。このような処理は並列処理に向いている。

並列処理の速度向上において一般に問題となる、データの転送量や転送回数について考えると、複数のデータに対して同一の処理を行なう場合、処理対象となるサンプルの量に比べて、処理の結果として得られるデータの量は桁違いに小さく、また、処理結果のデータ収集は計算終了後一括して行う場合が多い。よってデータ処理量(あるいはサンプル量)に比べてデータ転送量は遥かに小さい。本研究が目的とする、構造既知の全タンパク質の立体構造を解析する場合は、したがって、効率のよい並列処理が可能である。

1.4 統合化

上述の個々の要素技術は、統合化されていなければならない。統合化の方法として有効なのは、オブジェクト指向言語C++によるクラス、メソッドライブラリとして構築することである。

本システムは、上述のように、タンパク質の立体構造データベースPDBの自動読み取りモジュール、三次元幾何学解析のためのライブラリ、統計解析およびそのための行列処理ライブラリ、3次元画像表示ライブラリ、配列アライメントのための関数、並列化実行におけるデータ転送を簡単に実現できる関数群などを、C++言語のクラスおよびメソッドライブラリとしてまとめたものである。本システムを利用するユーザは、このライブラリを使って、簡潔にプログラミングをし、目的の解析を迅速に行なうことができる。

2. 各モジュール、ライブラリの詳細

この節では、PDBデータ読み取りモジュール、幾何学計算ライブラリ、統計処理ライブラリ(行列計算ライブラリ)、配列解析のライブラリ、三次元グラフィックスのための関数群、並列化のための関数群について詳細な説明を行なう。

2.1 PDBデータ読み取りモジュール

PDBデータ読み取りモジュールは、指定されたPDBデータエントリーのファイルを読み込み、それをpdb.dataクラスの構造化されたオブジェクトとしてメモリー上に置く。PDBファイルは、前述のように本来

の書式(文献²)にも書式の定義は存在する)を逸脱したもののや、エントリーごとに独自の書式拡張をしたものが多数存在する。そのため、この読み取り部分を作ることが非常に難しい。今回は、PDB Release 78の全ファイルが可読であることを目的として読み取りプログラムを作成した。その後、Release 80の読み取りを行なった結果、不可読なファイルは存在しなかった。よって、将来的に見て、ほぼ全ての標準書式を逸脱したファイルに対応できたものと考えられる。

一般的に電子化されたデータの質や形式が均質で無く、自動読み取りが難しい場合は、高速読み込みが可能で独自の中間データフォーマットを開発し、原データをこれに変換して利用することが多い。しかし後述のように、データ読みとりの部分を並列処理することで、読み取りに必要な処理時間は十数秒程度まで圧縮できることが判明したので、本システムでは、独自の中間データフォーマットを持つことはしなかった。

ファイルから読み込まれた `pdb.data` クラスのオブジェクトは、`protein` クラスのコンストラクタに引き渡された段階で、データ中の原子の座標の欠損などが検査され、かつタンパク質のアミノ酸残基ごとにデータをまとめ直すなどされて、`protein` クラスのオブジェクトが生成される。この検査と `protein` クラスのオブジェクト生成は、多くの処理時間を必要とする。PDBファイルの中でアミノ酸配列が書かれているセクションと、実際の原子座標が書かれているセクションとの配列の整合性を見るために、動的計画法を利用した比較も行なえる。

また、今後の分子動力学計算や精密な構造解析のために、一般のPDBでは欠損している水素原子についての情報を付加する機能も有している。水素原子付加においては、水素原子の位置座標の導出には、分子動力学計算の標準的なシステムである AMBER⁶⁾ のパラメータに基づいて行なうこととし、そのため、AMBERのパラメータファイルの読み取りも行なえるようになっていく。

2.2 三次元幾何学計算ライブラリ

幾何学計算ライブラリでは、まず、三次元空間での座標やベクトルを表す `vector3D` クラスを定義し、ついで、3次元空間での座標変換や回転を表す 3×3 行列 `matrix3D` クラスを定義し、これらの間の各種演算を定義している。更に、処理の効率化とプログラミングの便宜のために、空間の2点間の距離や、空間の与えられた3点で定義される角度、4点で定義される二面角を計算する関数も用意されている。また、三次元回転を表すための、`rotation` クラスを `matrix3D` クラスのサブクラスとして定義し、Euler角や、任意の方向の軸回りの任意角度によって定義される三次元回転を表す 3×3 行列を簡単に生成することができる。

タンパク質立体構造解析では、構造間の類似度評価のため、類似する二つの構造を、空間的に最適な位置

に重ね合わせる処理が頻繁に用いられる。これを簡単に実現できるようにするため、`vector3D` クラスの配列として、`conformation` クラスを定義した。このクラスでは、同じ数の位置座標から成る二つの `conformation` クラスのオブジェクトを最適重ね合わせ (Least RMSD Fitting) するメソッドが定義されている。

2.3 統計処理および行列計算ライブラリ

統計処理では、基本的な多次元量の平均、分散(共分散)を求める関数や、主成分分析を行なう関数群が定義されている。これらの計算の基盤となるのは、任意要素数の行列計算である。行列計算ライブラリは、統計処理以外の用途があることも考慮して、一般的な行列計算の機能(たとえば、転置行列や逆行列の計算など)も提供している。

なお、対象としている行列は、高々 100×100 程度の大きさであるから、行列計算そのものを並列処理により高速化することはここでは考えない。

2.4 配列解析のライブラリ

アミノ酸残基配列は、`pdb.data` クラスの中の `pdb.seqres` クラスのオブジェクトに格納される。このクラスでは、二つの残基配列間のアライメント(ペアワイズアライメント)を行なうメソッドを提供している。アライメントのためのアミノ酸残基間の類似度マトリックスとしては、一般的に知られているものから選べるようになっている。

2.5 グラフィック表示モジュール

グラフィック表示については、将来的には、広く流通している標準的な三次元物体記述言語で、立体構造を表示することが望ましい。しかし、現状では、この種の言語処理系で、とくにアニメーションの機能をもつものなどは、それほど広く普及したものが存在しない。そこで、新世代コンピュータ技術開発機構(ICOT)が開発した三次元グラフィック表示システム `3DView`⁷⁾ を用いることにした。よって、三次元物体記述言語は、`3DTalk` である。`protein` クラスおよび、`conformation` クラスは、自分自身の三次元の立体構造を表現する `3DTalk` のスクリプトを生成するメソッドを持つ。

2.6 並列化のための関数群

本システムでは、分散メモリ環境での並列プログラミングを念頭に置き並列化のために便利な機能を提供している。並列プログラミングのプロトコルとしては、MPIあるいは、MPC++(MTTL)⁸⁾ を用いることを前提としている。

分散環境での並列プログラミングで面倒なことは、複雑な構造をもつデータの転送と、異なるノードへの関数呼び出しである。そのうち異なるノードへの関数呼び出しについては、仮に本システムで関数呼び出しプロトコルを独自開発したとしても、ユーザにとってはその習得自身が難しいことを考慮し、本システムには組み込ま

れていない。MPIで異なるノードのオブジェクトに対して関数呼び出しを行なうときには、そのオブジェクトを同じノードに転送し、ローカル関数呼び出しするのがもっとも簡単な方法である。MPC++ を利用しているときには、同言語の `invoke`(同期呼びだし)、`ainvoke`(非同期呼びだし) を用いることで簡単に行なえる。

3. C++ 言語のクラス設計

この章では、各モジュール、ライブラリで定義されているクラスについて、そのデータ構造を示す。

3.1 pdb_data クラス

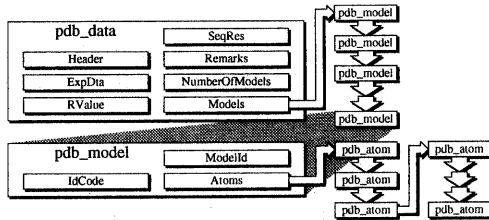


図1 pdb_data および pdb_model クラスのデータ構造

タンパク質の立体構造データベース PDB では、一つのタンパク質の立体構造は一つのファイルに書き込まれている。PDB のファイルでは、まず、そのファイルの記述するタンパク質が如何なるものであるかという説明 (HEADER, COMPND, SOURCE, AUTHOR, JRNL などのセクション) があり、次に、構造決定の実験方法 (EXPDTA セクション) などが書かれている。そして、そのタンパク質のアミノ酸残基配列が書かれており (SEQRES セクション)、その後、原子の座標を書いた部分 (ATOM セクション) がある。

NMR(核磁気共鳴) と呼ばれる立体構造決定法では、一般に実験で得られたデータから一意に構造を決定できない。この場合、1つの PDB ファイルの中に複数の推定される構造が書かれている。この場合、一つの推定される構造は、一つの MODEL として記述されている。そこで、PDB の一つのエントリーを表現する `pdb_data` クラスは、まず、その表現しているタンパク質についての説明に相当するフィールド、実験方法のフィールド、残基配列のフィールドに続いて、複数の構造の MODEL を `pdb_model` クラスのオブジェクトとしてもつようになっている。

次に、タンパク質は多くの場合、単独の鎖(分子)からなるのではなく、複数の鎖の複合体であることが多い。よって、一つの MODEL の中には、複数の鎖が書き込まれるようになっている。 `pdb_data` クラスのオブジェクトの生成時は、できるだけ PDB ファイルの内容を変更せずに読み取る必要があるので、この段階では、鎖ごとの分割などの細かい構造化は行なわず、構成する原子

の座標その他(温度因子や占有率など)を表すフィールドの配列にするに留まっている。

`pdb_data` クラスには、配列解析のためのペアワイズアライメントを行なうためのメソッドが用意されていて、タンパク質のアミノ酸残基配列間の相同性を計算することができる。

3.2 protein クラス

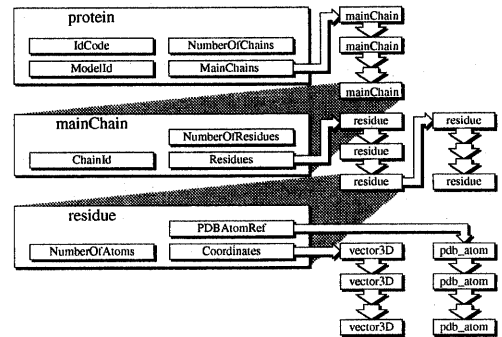


図2 protein および residue のデータ構造

上述の `pdb_data` クラスのオブジェクトの中の一つの MODEL(`pdb_model` クラスのオブジェクト) から生成される `protein` クラスのオブジェクトは、コンストラクタがこのオブジェクトを生成する段階で、ATOM セクションに記述されていた各原子を、まず鎖ごとに分けて `mainChain` クラスのオブジェクトに格納し、さらにそれをアミノ酸残基ごとにまとめて `residue` クラスのオブジェクトに格納する。すなわち `protein` クラスは、まず下部構造として、鎖一つ一つを表す `mainChain` クラスのデータ構造をもち、さらにそれぞれの `mainChain` クラスは、`residue` クラスのオブジェクトの配列をもつようになっている。

`residue` クラスのオブジェクトを作る段階で、PDB ファイルに書かれている原子座標の欠損や、アミノ酸残基そのものの欠損を検査し、欠損や矛盾があれば、標準エラー出力にレポートするようになっている。

また、`protein` クラスは、このクラスのオブジェクト以下でポインタ変数やリスト構造を用いて構造化されたデータを byte string に展開するメソッドを持っている。これによって、並列分散プログラミングにおけるデータ転送の高速化と簡便化を図っている。

さらに、分子動力学や、詳細な構造解析を行なうために、一般に PDB では省略されている水素原子を AMBER のパラメータを元に付加する機能も有している。

三次元立体表示を行なうために、このクラスでは、`3DView` が受けとる `3DTalk` 言語のスクリプトを生成する機能を持っている。これにより、タンパク質の立体構

造を容易に三次元グラフィックス表示できる。

3.3 三次元幾何学計算関連のデータ構造

vector3D クラスは、三次元空間でのベクトルを表す double 型のデータ 3 個からなるオブジェクトのクラスである。matrix3D クラスは、 3×3 行列を表すデータ構造で、double 型のデータ 9 個からなる。

conformation クラスは、任意個数の vector3D クラスのオブジェクトの配列をもつものである。conformation クラスのオブジェクトは、立体構造全体の回転や並進移動など vector3D クラスや matrix3D のオブジェクトとの間で、さまざまな演算ができるようになっている。conformation クラスのオブジェクト二つを三次元空間内で最適重ね合わせ (Least RMSD Fitting) する関数も用意されている。これは、二つのタンパク質立体構造の比較を行なうために必要な機能である。

conformation クラスは、この PAPIA System においては、タンパク質の鎖を表現していることが多いことから、conformation クラスには、記述されている構造を円筒で球を接続した三次元物体として 3DTalk 言語のスク립トで表現する関数があり、この形式で簡単に 3次元グラフィックス表示が実現できる。

3.4 行列計算、統計処理関連のデータ構造

vector クラスは、任意個数の double 型データを格納するオブジェクトのクラスである。また matrix クラスは、任意の大きさの行列を表すものである。おもに統計処理における主成分分析や主軸変換などに利用される。

統計処理のために dataSet クラス、multivariate クラス、histogram クラスが定義されている。dataSet クラスは、任意次元の多次元量を膨大に蓄えることのできるデータ構造である。multivariate クラスは dataSet クラスのサブクラスとして定義され、主成分分析を行なうためのメソッドが複数用意されている。このクラスでは、クラスター分析の一つとしてベクトル量子化を行なうメソッドも提供されている。これにより、複雑な分布をクラスター分類して表現することが可能である。histogram クラスは、multivariate クラスのサブクラスであり、データの多次元度数分布を求めるために用いられる。

4. 並列処理

本システムの開発目的は、タンパク質立体構造研究の効率化を狙ったものである。したがって、本システムの利用者である研究者が、なんらかの立体構造解析を行ないたいと考えたときに、以下の 2 点が重要である。

- (1) 目的の解析プログラムを迅速に、かつ簡単に書くことができる。
- (2) 膨大な処理を必要とする場合でも、プログラミングに時間を割かずに、効率のよい並列処理が可能である。

したがって、ここでの並列処理は、台数効果のみを上

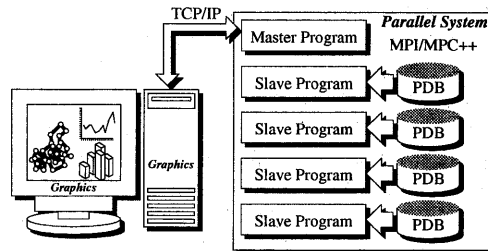


図 3 PAPIA System を用いた並列統合解析環境

げることを目的とするものではなく、プログラミングに必要な時間と、解析結果が得られるまでの時間の両方のバランスを考えて、ユーザーにとって、少ない努力で高性能の並列処理を可能とすることを目的としている。

PDB のエントリー数は 6000 を越える。そのため、タンパク質立体構造解析を全エントリーについて行なう処理の場合、単純なデータ並列処理を行なうだけで、高い台数効果が得られる。そこで、本システムではこのデータ並列の形式のプログラミングをユーザに奨励することとした。並列処理のために必要なプロトコルなどは提供せず、データ転送することが必要とされるデータ構造 (protein, conformation クラス) の byte string への展開の機能だけをライブラリはサポートしている。

5. 並列処理の実行効率

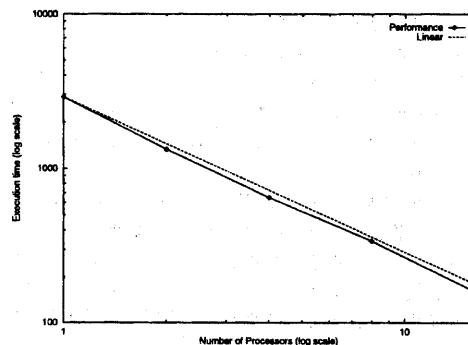


図 4 プロセッサ数 vs. PDB データ検査処理時間

並列処理の実行効率を調べるために、PDB のエントリーファイルを全て読みとり、一つ一つのエントリーの内容を厳しく検査するプログラムを作成しその実行効率を調査した。

一般的な単一 CPU のワークステーションでは、6000 を越える全エントリー (データサイズはおおよそ 2GB であり、メモリで構造化されるとおおよそ 3GB になる) を読み取ろうとする場合、メモリ不足になることが多い。また、各エントリーに対して原子の欠損などの詳細な検

査を行なうと、処理時間はおよそ単一CPUでは1時間程度必要である。本システム上で開発される解析プログラムは、起動時にPDBの全エントリーを読み込むものが多いと想定される。そのため、この部分の処理を並列化することは高性能化のために重要なことである。今回は、共有メモリ型のSPARCcenter2000E(主記憶5GB)を用い、MPIにより並列化したサンプルプログラムで、全エントリー読み取りとエントリー検査に要する時間を測定した。結果は図4の通りである。

CPU間相互でのデータ通信が全く行なわれないため、ほぼ完全な台数効果が得られている。よって、100台以上のCPUをもつ並列計算機においては、6000エントリーのPDBファイルの読み取りと検査を(ディスク入出力の速度で律速されることも考えられるが)十数秒で行なうことができ、実用上十分な速度と言える。

6. 考 察

現在PAPIA Systemは、共同研究者の間で利用され始めており、PDBの全エントリーから、解析対象となり得る質の高いデータの選択を行なうプログラムの開発や⁹⁾、タンパク質の立体構造中で、特定のアミノ酸残基ペアがどのような位置関係にあるのかを統計処理するプログラムなどで利用されている。

タンパク質立体構造解析に関するなんらかの研究目的があるときに、PAPIAに含まれるライブラリを利用すると、概ね数十行のプログラムを書くだけで、目的の解析を行なうプログラムを作成することができる。特に、行列計算や幾何学的計算のライブラリとの融合により、複雑な解析もかなり簡単に実現することができる。

並列システムの実行効率の面では、ほとんどの処理がデータ並列で実現できるため、ほぼ期待通りの並列効果が得られ、利用可能なCPU数に応じた速度の処理が可能である。

分子生物学も含めて、実際の応用プログラミングの中には、膨大なデータに対して同一の処理を行ない、統計処理するような場合が多く、これには、単純ながら、データ並列による並列プログラミングが非常に有効であると同時に、きわめて効率のよい並列計算環境を構築することができる。

このようなシステムのプログラミングは、一般的にデータの構造などに大きく依存し、また、必要に応じてさまざまな処理ができるように対応しなければならない。よって、必然的にデータベース管理モジュールと、解析に必要なさまざまなライブラリを構築し、それらを単純なプログラミングで繋ぎ合わせて統合化できるようなものになると考えられる。

実行効率のよいライブラリを作っておけば、並列化に関しては、単純なデータ並列であってもかなり効率のよい並列システムにすることができる。したがって、並列化を念頭においたデータベース管理や解析ライブラリな

どを統合したシステムは、今後重要な研究テーマになるものと思われる。

今後の展開としては、立体構造の物理的な解析、たとえば振動解析などのために、分子動力学計算を行なうモジュールも開発する予定である。現在すでにPDBデータでは一般的に書かれていない水素原子の座標なども付加する機能を持つので、物理的なポテンシャル計算の機能を追加することで、タンパク質のもつエネルギー計算も可能になる。これは、計算量がかなり大きいので、この部分を並列化することも考慮しなければならない。

参 考 文 献

- 1) Biosym/MSI 1995. "Insight II 95.0" 9685 Scranton Rd. San Diego, CA 92121-3752 USA. <http://www.msi.com/>
- 2) PDB release 80. 1997. "Protein Data Bank Contents Guide" included in *Brookhaven Protein Data Bank release 80.*, <http://www.pdb.bnl.gov/>
- 3) Bryant, S. H. 1989. "PKB: a program system and data base for analysis of protein structure." *Proteins* 5: 233-47
- 4) Chang, W; I. N. Shindyalov; C. Pu, P.E. Bourne 1994 "Design and application of PDBlib, a C++ macromolecular class library" *Comput. Appl. Biosci.* 10(6): 575-586
- 5) Hogue, C. W.V.; H. Ohkawa; and S. H. Bryant 1996. "A dynamic look at structures: WWW-Entrez and the Molecular Modeling Database". *Trends in Biochemical Sciences* 21: 226-229
- 6) Pearlman, D.A.; D.A. Case, J.W. Caldwell, W.S. Ross; T.E. Cheatham III; D.M. Ferguson; G.L. Seibel; U. Chandra Singh; P.K. Weiner; and P.A. Kollman 1995. "AMBER 4.1". University of California, San Francisco.
- 7) Akahoshi, M; K. Onizuka; M. Ishikawa; and K. Asai 1994. "A Three-Dimensional Animation System for Protein Folding Simulation". *Proc. of HICSS'94 Vol. of Biotechnology*: 173-182
- 8) Ishikawa, Y. 1997. "Multi Thread Template Library - MPC++ Ver.2.0 Level 0 Document -" *RWC Technical Report TR-96012*, <http://www.rwcp.or.jp/lab/mpslab/mpc++/mpc++.html>
- 9) Noguchi, T.; K. Onizuka; Y. Akiyama; and M. Saito 1997. "PDB-REPRDB, A Database of Representative Protein Chains in PDB (Protein Data Bank)" *Proc. of ISMB'97*: 214-217, The AAAI Press. <http://www.rwcp.or.jp/lab/pdappl/>