

B-017

Web サービスを用いたタンパク質立体構造予測システムの開発 Development of a Protein Tertiary Structure Prediction System using Web Service

廣安 知之[†] 三木 光範[†] 狩野 浩一[‡] 下坂 久司[‡]
Tomoyuki HIROYASU Mitsunori MIKI Koichi KANO Hisashi SIMOSAKA

1. はじめに

近年バイオインフォマティクス分野の研究が盛んに行われており、タンパク質の立体構造を予測することは、新薬の研究などに役立つ非常に重要な研究テーマである。しかし、タンパク質の立体構造をコンピュータシミュレーションにより最適化計算を用いて予測するためには、莫大な計算資源と高性能な最適器が必要である。また対象となるタンパク質も無数に存在している。

一方、近年 IT 資源をネットワークで結合し、統合して利用する Grid[1] が注目を集めている。ビジネス分野での Grid 技術の利用から、Web サービス標準技術 [2] を Grid に応用した Grid サービスが OGSA (Open Grid Services Architecture)[3] で定義され、OGSA の下位層に位置する OSGI (Open Grid Services Infrastructure) が Globus Metacomputing Toolkit により実装されている。また OSGI に代わり、新たに WSRF (WS-Resource Framework)[4] が提案されたことで、今後 Grid と Web サービスの標準技術が同一のものとなる。

このような背景から本研究では、Grid 環境における最適化計算システムの構築を目標とし、Web サービスを用いたタンパク質立体構造予測システムの開発について述べる。

2. 最適化計算システム

2.1 システム概要

提案システムとして、次探索点の決定を行う Optimizing Service と、解析計算を行い目的関数や制約条件の値を決定する Analyzing Service の 2 つから構成されるシステムを想定する。Optimizing Service において解析処理が必要になった際、Analyzing Service を呼び出すことにより解析結果を得て、最適化計算を実行する。各 Service は、複数のアプリケーションを連携させて作成することが考えられるため、提案システムでは次の機能が必要となる。

- Analyzing Service としてエンドユーザが任意のアプリケーション連携を構築できる機能
- 構築したアプリケーション連携を Optimizing Service の解析処理として利用できる機能

またこのようなシステムを構築する際、システムで利用される各アプリケーションの開発者は異なることが一般的であり、そのためアプリケーション間の情報交換はアプリケーションの入出力ファイルを用いて行うことが現実的である。

以上のことから、提案する Web サービスを用いたアプリケーション連携システムの概要を図 1 に示す。提案システムは、提案システムを用いて最適化計算システムの構築を行う Client、アプリケーションの実行やアプリ

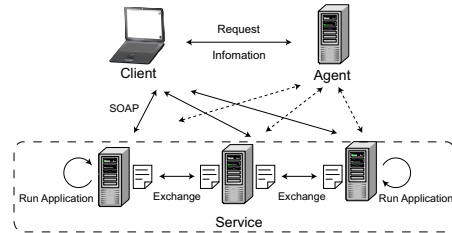


図 1: アプリケーション連携システム

ケーション間の情報交換のための機能を提供する Service、各 Service の情報を一元管理する Agent の 3 つの主要コンポーネントから構成される。

Client は、後述する Service の基本機能を適切な順序で実行し、アプリケーションの入出力ファイルを Service 間で相互に交換することによって、任意のアプリケーション連携システムを構築する。また同様に、構築した任意のアプリケーション連携を解析処理として利用するよう、Optimizing Service に指示することで、最適化計算システムの構築を行う。

2.2 基本機能

提案システムにおいて各 Service は、アプリケーションの実行やアプリケーション間で入出力ファイルを交換のために、4 つの基本機能を提供する。これらの基本機能は Web サービスの標準技術である SOAP-RPC[5] で実装されている。以下に 4 つの基本機能の詳細を示す。

1. Receive Files
SOAP-RPC の呼び出し側からのファイルを受信し、アプリケーションの入力ファイルとする。
2. Run Application
各 Service が管理するアプリケーションを実行する。
3. Return Files
SOAP-RPC の呼び出し側にアプリケーションの実行によって得られた出力ファイルを返信する。
4. Send Files
SOAP-RPC の呼び出し側が指定した他の Service に、アプリケーションの実行によって得られた出力ファイルを送信する。

2.3 設定ファイル

提案システムにおいて Client は、特定の書式に従った設定ファイルを記述することにより、アプリケーションの連携を指示する。設定ファイルに記述する内容を以下に示す。

- 実行する基本機能の総数 (NUMBER_OF_FUNCTIONS)
- 実行する基本機能の種類 (FUNCTION)
- 基本機能の呼び出し側の Service (FROM)
- 基本機能の呼び出し先の Service (TO)
- 送受信を行うファイルの数 (NUMBER_OF_FILES)
- 呼び出し側の Service が送受信するファイル名 (FROM_FILES)

[†]同志社大学 工学部
[‡]同志社大学大学院

- 呼び出し先の Service が送受信するファイル名 (TO_FILES)

2.4 Application Programming Interface

提案システムではアプリケーション開発者が容易にアプリケーション連携を利用し、新たなアプリケーションを開発するための API を提供する。Optimizing Service に属するアプリケーションでは、Coordinate クラスが提供する API を用い、Client が指定する任意のアプリケーション連携を解析処理として利用する。

```
int initialize(String service_info)
```

Service の初期化を行う。

```
int configure(String configfile)
```

設定ファイルを指定する。

```
int run()
```

アプリケーション連携を実行する。

3. タンパク質立体構造予測システムの構築

上記の様々な仕組みを用いてアプリケーションを複数連携させ、タンパク質立体構造予測システムの構築を行う。Client が提案システムを利用してタンパク質立体構造予測システムを構築するためには、以下のような手順で行う。

1. 現在利用可能な Service の情報を Agent から取得し、利用 Service を選択する。
2. 基本機能を組み合わせ、各 Service をどのように連携させるかを決定する。
3. アプリケーション連携の実行に必要なアプリケーションの入力ファイルを用意する。
4. アプリケーション連携を実行し、実行結果を取得する。

3.1 アプリケーションの選択

まず、エンドユーザは Agent からの情報をもとに連携を行うアプリケーションを選択する。

Analyzing Service

Analyzing Service として、タンパク質立体構造解析アプリケーション (Met-enkephalin) を選択する。Met-enkephalin は入力ファイルとして met.in を必要とし、出力ファイルとして met.out を出力する。met.in には解析を行うタンパク質を形成する二面角の角度、met.out には解析によって得られたタンパク質のエネルギー値が記述される。

Optimizing Service

Optimizing Service として、Simulated Annealing(SA) を選択する。SA は入力ファイルとして sa.config を必要とし、出力ファイルとして sa.out を出力する。sa.config には最適化の初期パラメータ、sa.out には最適化の結果が記述される。また、SA は解析処理を行う際に設計変数をファイル designVal に出力し、目的関数値を energyVal から入力する。この際、前述の API に設定ファイル (analyzer.config) を指定することで、designVal および energyVal を用いた任意のアプリケーション連携を可能とする。

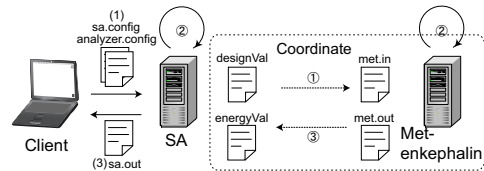


図 2: SA と Met-enkephalin の連携

\$NUMBER_OF_FUNCTION 3	\$FUNCTION 1 2 3	\$NUMBER_OF_FUNCTION 3	\$FUNCTION 1 2 3
\$FROM Client Client Client	\$TO SA SA SA	\$FROM SA SA SA	\$TO Met-e Met-e Met-e
\$NUMBER_OF_FILES 2 0 1	\$FROM_FILE saparameter.txt analyzer.txt	\$NUMBER_OF_FILES 1 0 1	\$FROM_FILE designVal
\$TO_FILE sa.config analyzer.config	\$FROM_FILE	\$TO_FILE met.in	\$FROM_FILE
\$FROM_FILE	\$TO_FILE saresult.txt	\$FROM_FILE energyVal	\$TO_FILE met.out
\$FROM_FILE saresult.txt	\$TO_FILE sa.out	\$FROM_FILE	\$TO_FILE

(a) 設定ファイル

(b) analyzer.config

図 3: 設定ファイル

3.2 ファイルの記述

次に選択したアプリケーションの連携方法を 2.3 節で述べた形式で設定ファイルに記述する。図 2 に示す構築例では、Client が最適化計算システムを実行するための設定ファイルと SA が Met-enkephalin との連携を行うための設定ファイル (analyzer.config) を記述する必要がある。図 3 にそれぞれの設定ファイルを示す。

3.3 必要なファイルの用意

SA の入力ファイルである sa.config を作成する。Met-enkephalin の入力ファイルである met.in は SA の出力ファイルである sa.out を入力ファイルとして用いるため、Client が用意する必要はない。

3.4 アプリケーション連携の実行

2.4 節で述べた API を用いて図 4 のような実行プログラムを作成し、アプリケーション連携を実行する。

```
public class Client(){
    public static void main(String args[]){
        Coordinate coordinate;

        public Client(){
            coordinate = new Coordinate();
        }

        coordinate.initialize(args[0]);
        coordinate.configure(args[1]);
        coordinate.run();
    }
}
```

図 4: Client プログラム

4. まとめ

本研究では、Web サービスを用いた汎用的なアプリケーション連携の仕組みを提案した。また、提案システムを用いてタンパク質立体構造予測システムの構築例を示した。

今後の課題は、Service の情報管理を行う Agent を実装し、Grid 環境で提案システムを実装、Globus Meta-computer Toolkit を用いたよりセキュアで信頼性の高いシステムの構築、Grid 計算環境における有効性の検討などがあげられる。

参考文献

- [1] Carl Kesselman Ian Foster. *The Grid : Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1998.
- [2] Web Service. <http://www.w3c.org/TR/xmlbase>
- [3] Jeffrey M.Nick Ian Foster, Carl Kesselman and Steven Tuecke. *DRAFT document : The Physiology of the Grid : An Open Grid Services Architecture for Distributed Systems Integration*. June 2002.
- [4] WS-Resource Framework. <http://www.globus.org/wsrf/>
- [5] SOAP-RPC. <http://www.w3.org/TR/2003/RFC-soap12-part2-20030624/>