

B-005

## 広域分散環境における最適化計算システムの開発 Development of an optimization problem solving system on the Grid

下坂 久司<sup>†</sup>  
Hisashi Shimosaka

廣安 知之<sup>‡</sup>  
Tomoyuki Hiroyasu

三木 光範<sup>‡</sup>  
Mitsunori Miki

### 1. はじめに

近年の計算機の処理速度の高速化、最適化手法の高性能化にともない、最適化の対象となる最適化問題は大規模・複雑化している。このような最適化問題を解くためには一般に、大規模な計算資源と高性能な最適化のためのアプリケーションが必要であり、これら計算資源と情報資源の統合的な利用が不可欠である。一方で近年、次世代の大規模な科学技術計算の基盤技術である Grid が高く注目されている [1]。Grid 技術の発達により、広域に点在する PC クラスタやスーパーコンピュータなどの計算資源、ストレージや大規模データベースなどの情報資源をシームレスに利用した、新しいアプリケーションの開発を行えることが期待されている [2]。

そのため、本研究では広域分散環境におけるアプリケーション連携システムを提案する。提案するシステムでは、広域分散環境においてアプリケーションを相互に連携させ、容易に最適化計算システムを構築できることを目標とする。

### 2. アプリケーション連携システム

#### 2.1 アプリケーション連携システムの概要

本論文で提案する、広域分散環境における最適化計算のためのアプリケーション連携システムは大きく、Client、Agent および Service 群の3つから構成されている。広域分散環境においてアプリケーションを連携させ、最適化計算システムの構築を行いたいエンドユーザは Client となる。また、広域分散環境上でアプリケーションの実行やアプリケーション間の情報交換のための機能を Service 群が提供し、それら各 Service の情報を Agent が管理する。

提案システムにおいて Client は、利用する Service を選択し、後述する Service の基本機能を適切な順序で実行することによって最適化計算システムを構築する。Service の基本機能はすべて Grid RPC [3] によって実装されており、Client は Service 群に対して複数の Grid RPC を実行する。また、ある Service から別の Service への情報交換も Grid RPC を利用して行われる。

#### 2.2 Service の基本機能

広域分散環境上の各 Service はアプリケーション連携を実現するために、Grid RPC の呼び出し側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 Receive Files 機能、Service が管理するアプリケーションを実行する Run Application 機能、アプリケーションの実行によって生成された出力ファイルを、Grid RPC 呼び出し側に返信する Return Files 機能、アプリケーションの実行によって生成された出力ファイルを、Grid RPC 呼び出し側が指定した他の Service に送信する Send Files 機

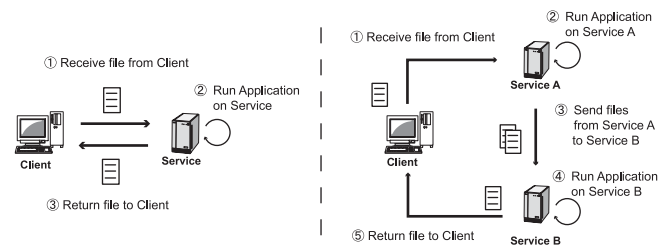


図 1: Examples of the integrated application system

能の4つの基本機能を提供する。このうち、Send Files 機能を除く3つの機能は1回の Grid RPC、Send Files 機能は2回の Grid RPC により実装される。

#### 2.3 アプリケーション連携の例

前節で述べた4つの基本機能を用いたアプリケーションの連携例を図1に示す。図1の左図は、Client が広域分散環境上のある Service を利用する最も単純な例、右図は Client が2つの Service 群を連携させて利用する例である。

左図の例において Client はまず、各 Service を管理する Agent から情報を取得し、利用する Service を選択する。そして次に、Service の入力ファイルを用意し、提案システムの設定ファイルに、利用する Service の指定と、Receive Files、Run Application、Return Files の3つの実行する機能を記述する。これにより、広域分散環境上の Service を利用し、アプリケーションの利用結果を得ることが可能となる。

また2つのアプリケーションを連携させる右図の例においても Client は同様に、Agent から取得した情報をもとに、利用する Service A および B を選択する。Service B の入力ファイルは Service A の出力ファイルを利用するため、Service A の入力ファイルのみを用意し、次の5つの処理を提案システムの設定ファイルに記述することで、アプリケーションの連携を実行できる。

1. Service A が Client から、入力ファイルを受信する (Receive Files)。
2. Service A が管理するアプリケーションを実行する (Run Application)。
3. (2) で生成された出力ファイルを Service B に送信する (Send Files)。
4. Service B が管理するアプリケーションを実行する (Run Application)。
5. Service B の出力ファイルを Client に返信する (Return Files)。

<sup>†</sup>同志社大学大学院工学研究科博士後期課程

<sup>‡</sup>同志社大学工学部

## 2.4 最適化計算システムの構築

次に提案システムを利用し、最適化計算システムを構築する。本論文で想定する最適化計算システムは、最適計算を行う最適化アプリケーションと、解析計算を行う解析アプリケーションを完全に分離したものを想定する。最適化アプリケーションでは、設計変数値に対応する目的関数および制約条件値が必要になった場合、アプリケーション利用者が指定した解析アプリケーションを実行し解析結果を得る。これにより、ある最適化アプリケーションは様々な解析アプリケーションに適用することが可能となる。図2に、最適化アプリケーションとして逐次2次計画法 (sqp)、解析アプリケーションとしてトラス構造物の解析アプリケーション (fem\_truss) を提案システムに登録し、最適化計算システムを構築した際のアプリケーション連携の例について示す。

この例で Client は、sqp および fem\_truss への初期設定ファイルの送信 (Receive Files)、sqp を実行することによる最適化計算の実行 (Run Application)、最適化計算終了後の sqp からの結果の返信 (Return Files) を行う。前節で述べたように、これらは4つの処理によるアプリケーション連携を、提案システムの設定ファイルに記述することで実現できる。また sqp は最適化計算の実行中に設計変数値に対応する解析結果が必要になった際には、「Analyzing for sqp」に記述されている3つの処理によるアプリケーション連携を fem\_truss に対して行い、解析結果を得る。このアプリケーション連携は、解析結果が必要な部分において最適化計算が終了するまで繰り返し実行される。

提案システムを利用して、図2の最適化計算システムを構築するために、Client はまず sqp に対して利用する解析アプリケーション (fem\_truss) を適切に指定する必要がある。さらに、sqp と fem\_truss 間のアプリケーション連携を指示する必要がある。提案システムにおいてこれは、Client が用意する提案システムの設定ファイルと同様の書式を用い、Client を sqp に置換した形で記述できる。適切に記述した設定ファイルは、最適化計算を実行する前に最適化アプリケーションへ送信しておき、最適化計算実行中に最適化アプリケーションが Client から送信された設定ファイルを読みこんで、解析処理に必要なアプリケーション連携を実行する。図2において Client から sqp に「opss.conf」というファイル名で、この設定ファイルが送信されていることがわかる。

## 3. 結論

本論文では、システムを利用するエンドユーザが広域分散環境に存在する複数の Service を選択し、それぞれを連携させることで容易に最適化計算システムを構築できるアプリケーション連携システムを提案した。また本論文では頁数の関係上記述できないが、簡単な情報ファイルを記述することで既存アプリケーションを容易に提案システムに登録できる仕組みや、提案システム上でのアプリケーション連携を利用する、新しいアプリケーションを作成するための Application Programming Interface(API) も提案システムは提供している。最適化アプリケーションはこの API の一部を用いることで、Client から指定された解析処理を行うことができる。さらに提案システム

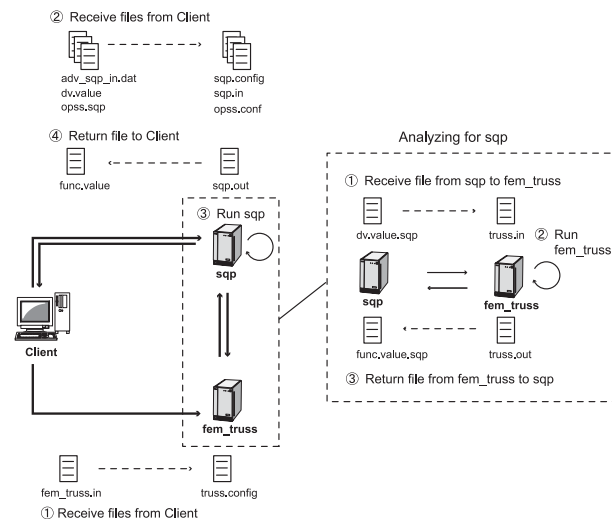


図2: Optimization system using the proposed system

において Client は、提案システムの設定ファイルを記述することが最も困難な作業となる。そのため、提案システムの設定ファイルを自動生成するツール (図3) も提案システムでは提供している。このツールでは、Client が広域分散環境上に存在する Service 間の情報交換を、視覚的に表現することで設定ファイルを生成できる。これらのことから、提案システムは、広域分散環境上で最適化計算システムを構築する上で非常に有用であるといえる。

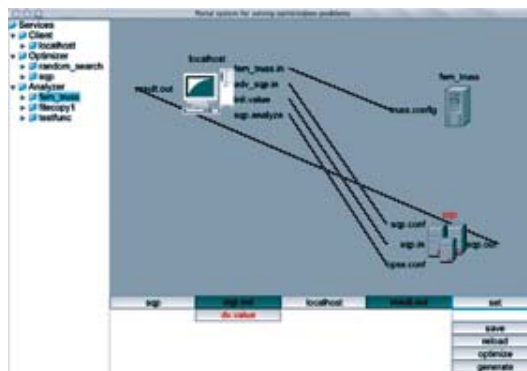


図3: User Interface

## 参考文献

- [1] I. Foster and C. Kesselman. *The Grid : Blueprint for a New Computing Infrastructure*. Morgan Kaufmann, 1998.
- [2] I. Foster, C. Kesselman, and S. Tuecke. The anatomy of the grid : Enabling scalable virtual organizations. *International Journal of Supercomputer Applications*, Vol. 15, No. 3, 2001.
- [3] K. Seymour, H. Nakada, S. Matsuoka, J. Dongarra, C. Lee, and H. Casanova. Grid rpc : A remote procedure call api for grid computing. *Technical Report ICL-UT 02-06, ICL, University of Tennessee*, 2002.