

分散環境に対応した数値シミュレーション可視化システム

3 Q-3

武宮博¹ 橋祐一¹ 布川博士² 野口正一³¹日立東北ソフトウェア ²東北大学電気通信研究所 ³東北大学応用情報学研究センター

1. はじめに

計算機の高性能化に伴い、これまで不可能であったより現実に近い数値シミュレーション、より大規模な数値シミュレーションが可能となってきた。

このようなシミュレーションでは、大量の計算データが生成されるため計算結果の解析においてデータをそのままの形で調べることが不可能になっている。そのため、シミュレーション結果の可視化の重要性が高まっている。

この可視化による結果解析を支援するシステムを構築する際の問題点としては、以下のものが挙げられる。

(1) 分散環境への対応

可視化による解析では、データ量の増加のために高い計算能力を必要とする処理が要求されると同時にUIに関するインタラクティブな処理も要求される。この二種類の処理を同一の計算機で行うのは無駄が多く、分散環境への対応が必要となる。

(2) GUI構築のフレキシビリティ

数値シミュレーションでは単に生成したデータを可視化するだけでなく、その可視化結果を元に新しいデータを作成しそれを可視化する(3次元データを任意断面で切断して2次元データを生成し、その面に沿った等高線図を出力する等)ことを繰り返して現象全体を解析していく。そのためユーザに高機能のUIを提供しその作業を支援しなければならない。

しかし、数値シミュレーションの対象分野は多岐にわたっており、種々の可視化手法が存在する。しかも新しい可視化手法が常に出て現れている。従って、分散環境で可視化による解析を行うことを前提にすると、UI部分の柔軟な変更や拡張への対応が必要とされる。

(3) データの再利用性の向上

可視化によって得られた結果は、多量のCPUパワーを利用して得られるにもかかわらず、再利用をサポートするシステムが提供されていない。

一般に、可視化による解析の過程では、多種のデー

タやイメージが生成される。これらを管理し、再利用しやすくする枠組が必要である。さらに、解析したいデータを簡単に提供してくれる枠組も必要である。

現在我々はこれらの問題点を解決するために、分散システムに対応した可視化シミュレーションシステムを構築中である。以下では、上記の問題に対する我々のアプローチ及びシステムの構成について述べる。

2. 構成モデル

第1章で述べた問題点を解決するために、我々は可視化システムを独立な4つのサブシステムに分割し、それらが連携して処理を行う構成にした(図1)。このシステムにより、第1章で述べた問題は以下のように解決される。

(1)の問題点に対応して、インタラクティブな処理を行うUI部と高い計算機能を要求されるデータ操作部を分割した。UI部は、画像の表示及びユーザとのインタラクションを行うサブシステムである。データ処理部は、原データを基にして新しいデータを生成するサブシステムである。

(2)の問題点に対応して、UIを独立なUIパーツの集合体として構成できるようにし、処理に応じて必要なUIパーツをUI部に供給するUIパーツ管理部を設定した。

(3)の問題点に対応して、データを管理し、必要に応じてデータ操作部にデータを提供するデータ管理部を設定した。

以下に、このシステムの利用の流れを説明する。

1. ユーザは希望する処理をUI部に要求する。
2. UI部はデータ管理部にユーザの要求を通知する。
3. データ管理部は必要なデータと、データ操作部をデータ操作部に転送する。
4. データ操作部は可視化に必要なデータ処理を行

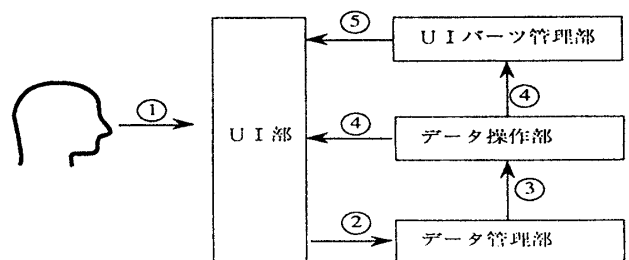


図1. モデル構成

い、結果をUI部に転送する。同時に、UIパーツ管理部にその可視化データに関する情報を転送する。

5. UIパーツ部は、データ操作部から受け取った情報をもとに必要なUIパーツをUI部に転送する。UI部は新しい可視化結果を描画する。同時に必要なUIパーツを表示する。

可視化による結果の解析はこのサイクルを繰り返すことによって進められる。

UIパーツ管理部は処理に応じて必要となるUIパーツをパーツ群の中から識別し、UI部に提供しなければならない。この識別のためにデータ操作部から渡される情報として、データ型を利用することにした。UIパーツは自分のView、どのデータの処理に対して必要とされるかに関する識別子(型名)、イベントを受け取った際のスクリプトを持つ。UIパーツ管理部は、データ操作部から処理されるべきデータの型を通知されると、UIパーツの識別子をもとに必要なUIパーツを選別し、UI部に転送する。

4. 流体計算可視化用の抽象データ型

現在我々は、3次元流体シミュレーション用にシステムを構築しており、表1のような型が存在している。型はそのデータに対して行われる操作が一体化した抽象データ型として取り扱われる。

操作は3種類に大別される。

- (1) その型内の変換(回転, 拡大縮小, 平衡移動など)
- (2) 新しいデータ型の生成(任意断面でのカット, 等値面の生成など)
- (3) 部分データの生成(ワイヤフレームデータの生成など)

操作はこの3種類のどれに該当するかによって、各サブシステムのどこで行われるかが定まっている。

(1) の場合は、操作はデータ管理部において行われ、その結果は直接UI部に送られる。

(2) の場合はUI部においてローカルに操作を行い、結果をそのまま表示する。

(3) の場合は、第2章の基本的な流れに沿って、データ操作部において行われる。

これにより、操作の性質にあわせた柔軟な処理が期

3次元データ用の型	2次元データ用の型	1次元データ用の型
3次元物理空間型	2次元物理空間型	1次元物理空間型
3次元計算空間型	2次元非構造物理空間型	1次元非構造物理空間型
3D流線型	2次元計算空間型	グラフ型
3Dワイヤフレーム型	Contour型	1D表型
等値面型	Filled島嶼図型	
3Dベクトル型	2Dワイヤフレーム型	
	2D流線型	
	2Dベクトル型	
	2D表型	

表1. システムで使用されるデータ型(抜粋)

待できる。

4. 分散可視化システム

上記のモデルに基づいたシステムを現在構築中である。システム構成は図2のようになっている。システムを記述するための言語としてはDeLis[1]を選択した。DeLisはLispに基づいた分散環境を構築するための言語で、分散システム内のサブシステム間の通信を記述できること、シェル機能やGUI構築のための関数を持っていることなどの特徴を持っており、上記のような分散システムの構築に適している。

DeLisによって記述された関数の集合体として分散数値シミュレーションシステム用言語を用意し、それによって各サブシステムの動作を記述している。

5. まとめ

本稿では分散環境に対応した数値シミュレーション用可視化システムを提案した。分散環境に対応することにより、利用者は分散システムを意識することなく、それぞれの処理に適した計算機上で処理を行うことができる。

現在このシステムは、科学計算の後処理のみに着目してシステムを構築しているが、科学計算自体との接続性や前処理に関しても考慮する必要がある。

参考文献

田布施, 三石, 宮崎, 野口: 分散環境記述のための言語系, 第45回情報全体(1992), 5-77

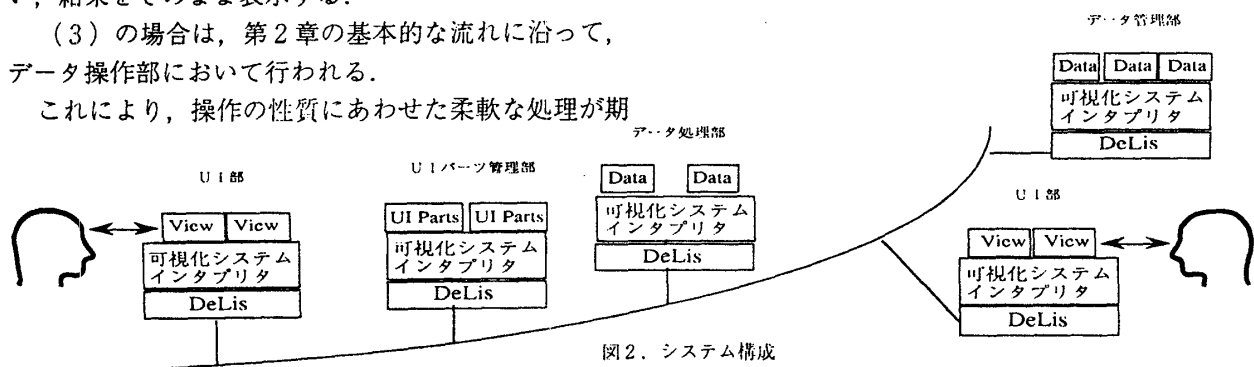


図2. システム構成