C107 大規模計算力学解析結果可視化のためのデータ処理

田村 善昭[○] (東洋大学), 古澤 寛行 (東都発条 (株)),

藤井 秀典 (東洋大学大学院), 宮本 勇輝 (東洋大学大学院)

Data Processing for Large-Scale Computational Mechanics Results

Yoshiaki TAMURA, Hiroyuki FURUSAWA, Hidenori FUJII and Yuki MIYAMOTO

ABSTRACT

Recent years, computational mechanics enlarges its scale owing to the progress of computer environment. Nowadays, we have wide variety of computer environment for computer simulations; from GRID computing to Earth Simulator. It is a great change. However, visualization process has not been changed. Whole of computed results are once loaded on the memory of PC and then visualized. This model of visualization process is no longer valid because the data size would exceed the memory of PC or the data be divided and distributed. In the present paper, data handling methodology for the postprocess of the large-scale computational mechanics results are discussed. Data format and data transfer timing from the remote computing server to the local PC are focused with two different data structures, namely, structured and unstructured grid data.

Keywords: Visualization, Large-scale simulation, Postprocess, Data handling, Computational mechanics

1. はじめに

計算機の日進月歩の進歩により、計算力学による解析はますます大規模化してきている。このことは計算力学が本格化した1970年代からおそらく変わっていないが、そこで用いられる計算機やその環境は大きく変化を遂げている。従来は、大きな演算機、俗にスーパーコンピュータと呼ばれる特殊な計算機があって、計算はバッチジョブという形で順に処理され、得られた結果はハードディスクに格納された後、ftp等で手元のワークステーションなどに転送され、後処理(可視化など)が行われる、というのが典型的な形態であった。しかし、単一のCPUの演算性能の向上に限りが見られるようになった1990年代あたりから、複数のCPUを用いる並列計算機が演算機の主流となり、今日ではその形態も複雑化し、グリッドコンピューティングのように、単にネットで繋がっているだけの複数台の計算機を利用して規模を大きくするものから、PCを数台から数十台あるいはそれ以上接続して用いるPCクラスター、1つのチップに複数のCPUを搭載するマルチコア化されたMPUを1つのボード上に複数並べてメモリーを共用化する演算機、あるいは地球シミュレータのようにかつてのスーパーコンピュータが1つのボードに収められてそれが数百台あるいはそれ以上並んだものなど様々である。これに対して、後処理、特に可視化については、一対の要素データ(格子データ)と結果データ(1つのファイルにまとめられている場合もあるが)を読み込んで可視化するという手順は

基本的には変わっていない。使用される演算機、CPUが複数になったことで、結果のファイルもそれと同じ数に分割されていたり、置かれている場所も必ずしも1カ所ではなかったり、逆にまとめられた結果、ファイルサイズが極端に大きくなっているなど、これまでの手順で処理するには不都合なことが多くなっている。そこで、大規模化と共に変化している解析に対応して、可視化の手順も変化が求められていると言える。

これらについては、いくつかの先行研究がある。白井ら¹⁾は分散保持されたデータに対して、それと対応する(並列)計算機を援用することで大規模な可視化を可能にしている。武井ら²⁾は、演算に用いた計算機で可視化処理を行い、画像データのみを転送する手法を提案している。また、上沢ら³⁾は、データを演算機で加工してより規模の小さいもの(具体的にはボリュームデータ)とし、転送して用いるアイデアを提案している。

ここでは、可視化の手法ではなく、可視化に至るまでのデータの流れについて議論する。特に、解析が構造的な要素(構造格子、直交等間隔格子などと呼ばれるもの、主に差分法で用いられる)のデータと、構造的でない要素(非構造格子などと呼ばれるもの、主に有限要素法などで用いられる)のデータについて検討する。

2. 可視化システム

2.1 可視化システムにおけるデータの流れ

まず、解析結果を可視化する際のデータの流れを整理してみると、Fig. 1のようになる。まず、要素データと

結果データを何段階か（0段階もありうるが）加工する。加工には、データを減らすための間引きや分割されている（置かれている場所そのものが分散している場合も含む）ファイルの統合などが考えられる。次に、可視化する手法（等高線表示、カラーの面塗りその他）に応じて処理を行い、等高線や面の幾何データを取り出す。この作業をここでは可視化処理と呼ぶ。可視化手法や可視化処理には様々なものがあるがここでは特に議論しない。結果としては、どのような解析結果からはじめてもここでは線分や多角形などの幾何形状とそれに付随する色や透明度などの情報（属性）がデータとして用意される。次に、この幾何データを元にコンピュータ・グラフィックス（CG）によって画像を作る処理をCG処理と呼ぶことにする。CG処理の結果得られるのが最終的に我々が見る可視化画像である。CG処理においてもその途中でデータが取り出される場合がある。メタファイルなどと呼ばれるものがこれに当たる。このように、可視化においてはいくつかの段階で異なった形式のデータが作られる。データ処理の観点から、これまでに提案されている大規模可視化システムのほとんどは、Fig. 1の①から⑤のどの段階でデータを演算機から手元の可視化用PCに転送しているかで分類できる。

Fig. 1 Visualization process and intermediate data

①は元の生データ、②は間引くなど加工しているが、それ自体で完結している（すべての情報を残している）データ、③は可視化処理を終えた幾何データ（CGデータ）、④はCG処理の途中で生成されるデータ（メタファイルなど）、⑤は最終的に得られる画像である。例えば、文献1)は③ないし④、文献2)は⑤、文献3)は②のデータを元の演算機から手元のPCに送っている。

つぎにこれらの利点・欠点を考察する。手元に近い順に考える。まず、最終画像⑤を送る方法は、画像1枚当たりのデータ量が小さい（圧縮すれば数十KBから数百KBの下の方）ことが利点であるが、会話的な視点変更などを行うには応答性が低い。どちらかという、あらかじめ表示する内容を決めておく使い方になるかと思

れる。可視化に要する演算機の負荷も問題になるかも知れない。つぎに、CGの中間ファイル④を送るのは、例えばAVSなどにこれに近いシステムがあるが、リモートの演算機と手元のPCとが密に連携をとる必要がある。

AVSでは関係するすべてのマシンでAVSを起動することになる。環境が許せば有効な方法の1つである。幾何データ③を転送するのは分かりやすい方法である。リモートの演算機と手元のPCでの役割分担もはっきりする。ただし、リモートの演算機で可視化処理のすべてを行うため、それなりの負荷が掛かる。また、表示する内容（線や面の形状の他、色などの属性値も含めて）が変わるたびにデータ転送が必要となるので、全体として転送するデータは少なくないと思われる。これもリモートの演算機が自由に使えれば可能な方法である。著者らが提案するのは、要素データと結果データを加工したデータ②を転送する方法である。加工はするが、可視化処理を施す前なので、手元のPCでは従来と同じように送られたデータに対して可視化処理を行うことになる。この方法の利点は、手元のPCで動作する可視化ソフトウェアが従来から余り大きな変更を要求されない、つまり、データがすべて手元にある場合と近い感覚で用いることができることである。リモートの演算機はデータを加工する程度の負荷でよい。問題はどこまでデータを削減できると、データを転送する頻度である。著者らはここに焦点を絞っている。次節では著者らの具体的なデータ削減方法について論じる。

2.2 本研究のデータ処理

著者らの考え方は、なんらかの方法で、要素データと結果データのサブセットをリモートの演算機上で用意し、これを手元のPCに転送して可視化するというものである。ここでいうサブセットとは、それらのデータだけで過不足なく、あらゆる関数（圧力、速度、応力、…）について可視化ができることを意味する。例えば、全体の中からある領域の要素データを取り出したとして、対応する結果データのうち、圧力以外のものは捨ててしまう、ということではなく、その領域についてはすべての結果を保持する、という意味である。従って、その領域をどのように選ぶかがポイントとなる。

要素データから領域を選ぶとなると、要素データの形式が問題となってくる。先に述べた構造的である。構造的があって、描く場所が限られる場合（例えばある特定の面上など）は、その場所のみデータを取り出せばよい。これは多くの場合、多次元配列から規則的にデータを取り出すことで実現できる。これには、Vistraceというソフトウェア⁴⁾が利用できる。Vistraceは計算機のメモリーもしくはハードディスクのファイルから規則的に並んだデータを他の計算機に送るライブラリである。これを用いて実現した可視化例がFig. 2である⁵⁾。ここでは1024×1024×1024という大規模な超音波伝播解析結果の2

つの面を表示しているが、領域は 16 分割されていて、各領域の要素データが約 0.8GB、結果データが約 1.3GB であるのに対して、手元の PC で使用したメモリーはわずかに 65MB である。

Fig. 2 Visualization example of very large-scale simulation

構造的な場合、あるいは構造的があっても、データのどの部分を転送すればよいかあらかじめ分からない場合は、なんらかの処理が必要になる。単に空間配置の問題であれば、要素データを調べることで必要な領域が特定できるため、リモートの演算機で処理することも考えられる。この例が Fig. 3 である。これは、円管のき裂解析結果であるが、Fig. 3 (a) が全体を表示しているのに対し、Fig. 3 (b) はき裂の付近のみを抽出している。この例では、き裂付近は極めて細かい要素が用いられているため、き裂付近に限ってもそれほどデータ量は削減されない (1/2 程度) が、方法論としてリモートの演算機で処理することは十分に現実的であると考えられる。一方、結果データも合わせて参照しないと領域が確定しない場合、例えば、等値面のような可視化では、全領域の探索が必要である。これを演算機で行ってもよいが、これはいわば可視化処理の一部であり、可視化処理まで演算機でする、というモデルになってしまう。その可能性を排除する必要はないが、ここでは可視化処理はすべて手元の PC で行うという方針でいくとした。具体的には、全く何も情報がない状態では、すべてのデータを探索せざるを得ないので、適当にデータを分割し、転送しては逐次可視化していく。このときに、分割した領域のどこに必要なデータがあったか、より詳細には、分割した領域のさらにどの部分が対象であったかなどを可視化の履歴と共に記録する。すると、2 回目はその情報を利用することで、転送するデータを減らしたり、可視化の

時間を短縮できる可能性がある。3 回目以降は、更に情報が増えるので、より高速化が期待できる。このアイデアの実装例は文献 7) に示されている。

(a) Whole elements visualized

(b) Part of elements visualized

Fig. 3 Crack simulation on a straight pipe

3. 考察

ここで示した例はデータの規模として、 $512 \times 512 \times 512$ あるいは $1024 \times 1024 \times 1024$ 程度までである。これよりデータが大きくなると、例えばそれは次期スーパーコンピュータなどを想定するものであるが、ここでの処理形態は適切でないかも知れない。現在や少し以前における地球シミュレータのように、その時代その時代で突出した性能を持つスーパーコンピュータは存在するが、それを利用できるのはひとにぎりの利用者であり、そこ

までを含めた汎用的な可視化システムを考えることは余り効率的でなく、それらの突出した計算機にはまた特別な可視化があつてしかるべきであろう。これらについては、いくつかの先行研究があるが、グラフィックスクラスターのような可視化専用並列計算機が1つの答えではないかと考える。本研究の対象はそれよりも少し規模の小さいものである。

4. まとめ

大規模な解析結果に対する可視化について、データの処理を中心に議論した。ここで提案したデータ処理については、今後順次実装していく予定である。

参 考 文 献

- 1) 白井出, 吉村忍, 矢川元基: 没入型仮想現実空間における大規模有限要素解析結果の並列可視化, 日本機械学会, Vol. 65, No. 638 (1999), pp. 2017-2023
- 2) 武井利文, 松本秀樹, 土肥俊: 大規模非定常数値シミュレーションのリアルタイム可視化——並列計算サーバによる可視化方式の実用化に向けて, 情報処理学会, Vol. 41, No. SIG 8 (2000), pp107-118
- 3) 上沢和真, 江原康生, 小山田耕二, 酒井晃二, 土井章男: 遠隔地間における大規模データのリモート可視化の効率化に関する検討, 電子情報通信学会, CQ2002-143, MVE2002-117 (2003), pp. 75-80
- 4) 田村善昭, 小笠温滋, 中島拓之, 藤井孝蔵: ライブラリス・リアルタイム可視化システムとその利用, 第 33 回可視化情報シンポジウム公園論文集, 可視化情報学会 25-Suppl. No.1 (2005), pp. 251-254
- 5) H. Furusawa, Y. Tamura, "A New Concept of Visualization for Large-Scale Simulations," APCOM'07 - EPMESC XI, Kyoto, Japan, Dec. 2007
- 6) 宮本勇輝, 田村善昭: 大規模数値計算のための可視化処理, 第 37 回可視化情報シンポジウム, 可視化情報学会, (2009) (発表予定)
- 7) 藤井秀典, 古澤寛行, 田村善昭: 大規模解析結果の可視化処理の改良, 第 37 回可視化情報シンポジウム, 可視化情報学会, (2009) (発表予定)