8F13

高精度な大規模細胞流動計算のための GPU 計算基盤の開発

Development of a GPU computing platform for accurate simulations of large-scale cellular flow

正 ○今井陽介 (東北大学) 学 松永大樹 (東北大学)
正 石川拓司 (東北大学) 正 山口隆美 (東北大学)Yohsuke IMAI, Department of Bioengineering and Robotics, Tohoku University, 6-6-01 Aramaki Aza Aoba, Sendai,
Daiki MATSUNAGA, Tohoku University, Takuji ISHIKAWA, Tohoku University, Takami YAMAGUCHI, Tohoku
University

Key words: Cellular flow, Capsule suspension, Boundary integral method, Multi-GPU

1. 緒言

循環器系では、体積率 20%~45%程度の赤血球がサスペンションとして流動している。赤血球は膜と内部流体からなるシンプルな細胞であり、カプセルとしてモデル化できる。境界積分法⁽¹⁾は膜面での応力のジャンプ条件を厳密に記述できる手法であり、球形カプセルの単体挙動⁽²⁾や二体カプセルの相互作用⁽³⁾の解析などが報告されている。一方で計算負荷が高いため、少なくとも数百から数千のカプセルを同時に計算する必要がある濃厚サスペンションの解析には至っていない。我々は、ハイパフォーマンスコンピューティングの分野で注目を集める GPU 計算を導入し、1000 カプセル規模の計算を可能にしたのでこれを報告する。

2. 方法

2. 1 計算手法

Stokes 流れ場を流動する N 個のカプセルを考える。流れ場の粘度を μ とし、簡単のため、カプセル内部流体との粘度比を 1 とする。膜における Cauchy 応力テンソルは、変形勾配テンソル $\underline{E}$ 、歪みエネルギー関数 w 、および Green-Lagrange の歪みテンソル $\underline{e}$ より以下の式で与えられる。

$$\underline{T} = \frac{1}{J_s} \underline{F}_s \cdot \frac{\partial w}{\partial \underline{e}} \cdot \underline{F}_s^T \quad (1)$$

ここで $J_s = \det(\underline{F}_s)$ である。仮想仕事の原理を用いれば、平衡方程式を以下のように記述できる。

$$\int_{S_c} \hat{\underline{u}} \cdot \underline{q} dS = \int_{S_c} \hat{\underline{\varepsilon}} : \underline{T} dS \quad (2)$$

ここで S_c はあるカプセル C の表面、 $\hat{\underline{u}}$ は仮想変位、 $\hat{\underline{\varepsilon}}$ は仮想歪みテンソル、 $\underline{q} = [\underline{\sigma}] \cdot \underline{n}$ は膜に作用する応力のジャンプである。このとき膜の速度は以下の境界積分方程式により与えられる。

$$\underline{v}(\underline{x}) = \underline{v}^\infty(\underline{x}) - \frac{1}{8\pi\mu} \sum_N \int_{S_c} \underline{J}(\underline{x}, \underline{y}) \cdot \underline{q} dS(\underline{y}) \quad (3)$$

ここで $\underline{v}^\infty$ はカプセルが存在しない場合の流速、 $\underline{J}$ は Oseen テンソルである。

式(1)から(3)を解くための計算手法として、Walter らが提案したもの⁽⁴⁾を用いる。詳細は参考文献に譲るが、有限要素法で解かれる式(2)と境界積分法で解かれる式(3)をカップルする手法である。

2. 2 GPU 計算

Stokes 流れ場を流動する N 個のカプセルを考える。流れ GPU 計算の実装は CUDA を用いた。あるカプセルを M 個の node からなる非構造格子に分割すれば、式(1)、(2)より $3M$ の q を未知数とする有限要素マトリクスを得る。式(2)の右辺は element 単位の積分の重ね合わせであるので、element 単位の thread によって並列に実行する。有限要素マトリクスは疎行列の連立一次方程式であり、node 単位の thread を用いた疎行列の行列解法によって解く。ここでは BiCGStab 法を用いた。式(2)から分かるように、この有限要素マトリクスはカプセルごとに閉じており、カプセルの数が増えた場合にも、連立一次方程式のセットが増えるだけである。最後に有限要素法で得られた q を式(3)に代入し、Gauss 積分を用いて積分することで速度 v を得る。これについても node 単位の thread を用いる。これらにより式(1)から式(3)までの全ての過程を完全に GPU 計算に実装することができる。

さらに、マルチ GPU を用いた並列計算を行うため、MPI ライブラリを用いた並列計算アルゴリズムを構築した。一つの GPU が少なくとも一つ以上のカプセルの計算を担当するものとすれば、式(1)、(2)はカプセルごとに閉じた計算であるため、複数 GPU の場合にも通信の必要はない。一方、式(3)は全てのカプセルの node の q を必要とする。マルチ GPU 計算ではデバイスメモリ上のデータをメインメモリに転送し、MPI によって通信した後、再度、メインメモリからデバイスメモリに転送するため、通信工程が計算のボトルネックとなりやすい。ここでは、ある被積分領域に対する積分計算と、次に被積分領域となる領域のデータ通信を同時に行うことで通信時間を隠蔽した。

3. 結果と考察

単一の GPU による計算の演算性能を NVIDIA GeForce GTX580 を用いて計測した。図 1 は、単純せん断流れ場を流動するカプセル挙動の計算について、総 node 数に対する演算性能を示したものである。ここで一つのカプセルは 2562 node で構成している。なお式(3)の積分計算が全体の計算時間のほとんどを占めており、カプセルの格子解像度を変化させた場合にも総 node 数が同じであれば、演算性能はほぼ同

じになる。node 数が 10^4 (カプセル 16 個) を超えると、演算性能はほぼ 1TFLOPS となる。本計算手法ではメモリアクセス速度に比べ演算速度が支配的であり、さらに密行列演算であることからキャッシュのヒット率が非常に高いが、Shared memory を利用した場合、10%程度、演算性能に改善がみられ、1.15TFLOPS に達した。なお Intel Core i7-930 を用いた CPU 1core の計算では、約 2.5GFLOPS 程度の演算性能であり、最適化等が不十分であるためあくまで参考値ではあるが、GPU の実装により約 500 倍の加速が達成された。

Fig.1 Single GPU performance on a NVIDIA GeForce GTX 580. Single CPU core performance on an Intel Core i7-930 is also shown as a reference.

マルチ GPU の演算性能を計測するため、NVIDIA GeForce GTX580 を搭載したパーソナルコンピュータを QDR infiniband により相互接続した計算機クラスターを用いた。図 2 は GPU の数に対する演算性能の変化を示したものであり、1GPU あたりの node 数を 2562, 10248, 40992, 163968 とそれぞれ固定した結果である。直線は単一 GPU のときの演算性能から見積もられる理想性能であり、全ての結果において直線と重なっていることから、通信の隠蔽によって、並列台数に対する完全な弱スケーリングが得られることが分かる。8GPU のときの演算性能は 9TFLOPS を超える。

図 3 は 1000 カプセルの流動を計算した例である。8GPU の並列計算で約 10 日の計算時間となる。仮に 4 コアの CPU を 8 台並列で計算したとすると約 4 年の計算時間がかかり、あるいは 10 日で計算を終えるには、1000CPU 以上の並列計算が必要となる。

4. 結言

高精度な大規模細胞流動計算のための GPU 計算手法を開発した。固体膜に対する有限要素法、流体に対する境界積分法を GPU に完全実装し、シングル GPU の計算では 1.15TFLOPS の演算性能を得た。これは CPU 1 core と比較して約 500 倍の高速化である。さらにマルチ GPU 計算に対応するため、MPI ライブラリを用いた並列計算コードに拡張し、通信時間を隠蔽するアルゴリズムによって、8 台までの並列計算に対して完全な弱スケーリングを得た。8GPU の小規模 GPU クラスターでも 1000 カプセル程度の流動計算が可能であ

り、したがって従来行われてきた単一カプセルや二体カプセルの解析から十分な発展が期待できる。なおここでは外部流体とカプセル内部流体の粘度比を 1 とし、式(3)を単純化した。粘度比が 1 でない場合についても同様の加速が得られることを確認している。

Fig.2 Multi GPU performance on a NVIDIA GeForce GTX 580 cluster. Lines show ideal speed-up for parallel computation.

Fig.3 An example of simulations of capsule suspension. Dynamics of 1000 capsules are solved. Color represents tension distribution on the membrane.

参考文献

- (1) C. Pozrikidis, Boundary Integral and Singularity Methods for Linearized Viscous Flow (1992) Cambridge University Press.
- (2) E. Lac, D. Barthès-Biesel, N.A. Pelekasis, J. Tsamopoulos, J. Fluid Mech. 516 (2004) 303-334.
- (3) E. Lac, A. Morel, D. Barthès-Biesel, J. Fluid Mech. 573 (2007) 149-169.
- (4) J. Walter, A.-V. Salsac, D. Barthès-Biesel, P. Le Tallec, Int. J. Numer. Meth. Eng. 83 (2010) 829-850.