

適合細分化格子を用いた格子ボルツマン法による非球形固体粒子との直接相互作用計算に基づいた大規模混相流シミュレーション



研究背景と目的

固体粒子を多数含んだ混相流は化学プラントなどの流動層で利用され産業上非常に重要な役割を果たしているが、その流体力学的振る舞いは十分に理解されていない。サスペンション・フローやデブリス・フローなど、粒子密度や粒子サイズに応じて流れが大きく変化する。これまでは経験則に基づいたモデル式が使われてきたが、粒子間衝突や非球形の粒子形状を考慮することができないため、現実の複雑な流体挙動の解析が十分できないという問題点がある。必然的に大規模計算となるため、非球形固体粒子との直接相互作用計算に基づいた大規模混相流の実問題に対するシミュレーションをエクサスケールで実現するための計算手法の開発と、性能/コストの高いGPUマシンでの実証を目的とする。

研究計画

メモリの連続アドレスが確保できる等間隔直交格子(構造格子)を用い、非圧縮性流体計算に対して疎行列計算を行う半陰解法ではなく、弱圧縮性を許す完全陽解法を用いる。計算手法として、単相非圧縮性流体について十分検証がなされている**格子ボルツマン法**を用いる。実問題への適用を考えたGPUマシンに適した適合細分化格子によるアプリケーションを開発する。

- D3Q27型の格子ボルツマン法に MRT 時間積分を導入する。複雑形状の境界条件には境界埋込み法ではなく、Interpolated Bounce Back 境界を用いる。
- 境界近傍を高解像度で計算するために、8分木による適合格子細分化法を導入し、領域分割にはHilbert空間充填曲線を用いる。リーフにはGPU計算に適したサイズのパッチを当てる。
- 複雑形状の固体粒子と流体の相互作用には、Immersed Moving Methodを用いるが、さらなる移動境界条件の性能向上のために新しい手法の開発を検討する。

研究グループ

代表者	青木 尊之	(東京工業大学)
副代表者	小林 宏充	(慶応義塾大学)
課題参加者	須賀 一彦	(大阪府立大学)
課題参加者	桑田 祐丞	(東京理科大学)
課題参加者	森口 周二	(東北大学)
課題参加者	長谷川 雄太	(東京工業大学)
課題参加者	渡辺 勢也	(東京工業大学)

MRT(Multiple Relaxation Time)モデル

格子ボルツマン法の時間積分に、これまで多く用いられてきたSRTではなく、MRT (Multiple Relaxation Time) を用いることにより、計算の安定性を大幅に向上させる。

$$S = \text{diag}(0, 0, 0, 0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}, s_{11}, s_{12}, s_{13}, s_{14}, s_{15}, s_{16}, s_{17}, s_{18}, s_{19}, s_{20}, s_{21}, s_{22}, s_{23}, s_{24}, s_{25}, s_{26}, s_{27})$$

$$s_1 = s_2 = \frac{1}{\tau}$$

$$s_3 = 1.54, s_4 = 1.5, s_{11} = 1.83, s_{16} = 1.4,$$

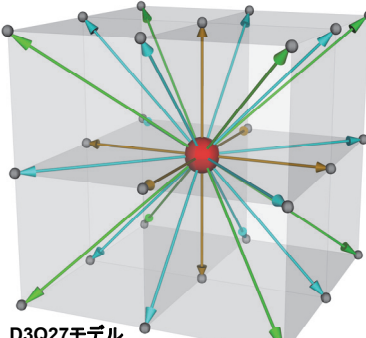
$$s_{17} = 1.61, s_{18} = s_{20} = 1.98, s_{23} = s_{26} = 1.74$$

$$\text{衝突項} = M^{-1} S M (f(x, t) - f^{eq}(x, t))$$

K. Suga, Y. Kuwata, et al.: A D3Q27 multiple-relaxation time lattice Boltzmann method for turbulent flows. Computers and Mathematics with Applications, Vol. 69, No. 6, pp. 518–529 (2015).

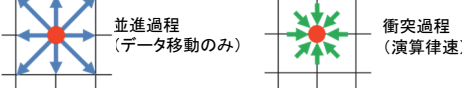
格子ボルツマン法

格子ボルツマン法は、流体をいくつかの離散速度を持った仮想粒子の分布関数と仮定し、位相空間で並進と衝突による分布関数の時間発展を計算する手法である。分布関数の速度には、1タイムステップで粒子がちょうど隣接格子点に移動するような速度のみを選び、離散化する。速度空間は複雑境界条件に対して最も精度よく表現できる3次元27速度(D3Q27)モデルを用いる。



$$|f(x + c_\alpha \Delta t, t + \Delta t) = |f(x, t) - \frac{1}{\tau} (|f(x, t) - |f^{eq}(x, t))$$

$$f^{eq}(x, t) = w_\alpha p \left(1 + \frac{c_\alpha \cdot u}{c_s^2} + \frac{(c_\alpha \cdot u)^2 - c_s^2 |u|^2}{2c_s^4} \right)$$



Immersed Moving Method

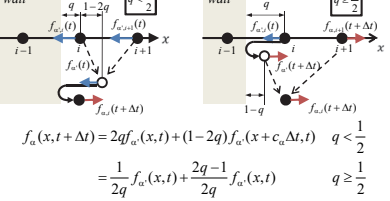
各計算セルの個体体積占有率に基づいて、流体と固体の重み付けを行い、移動境界条件を課す。

$$\beta_n = \frac{\text{体積}}{\text{セル体積}}$$

$$f(x + c_\alpha \Delta t, t + \Delta t) - f(x, t) = -(I - \beta_n) M^{-1} S (m(x, t) - m^{eq}(x, t)) + \beta_n f^{eq}$$

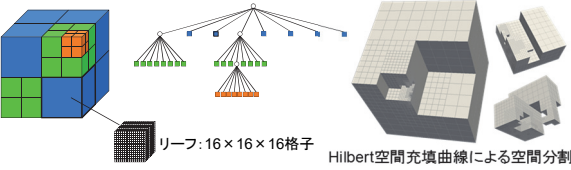
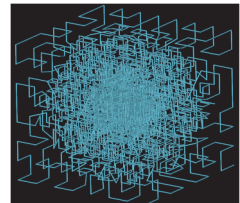
Interpolated Bounce-back

速度分布関数を補間し、格子点間にある任意の位置の壁に対して、2次精度で境界条件を課す。



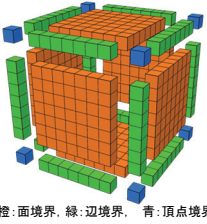
8分木に基づいた適合細分化格子

計算空間の中で層流から乱流への遷移する場所や物体近傍で境界層が剥離する場所などでは他より圧倒的に高解像度で計算する必要がある。直交格子を8分木データ構造に基づき再帰的に細分化する適合細分化格子を導入する。GPU計算に適した構造とするため、計算が行われるリーフを16×16×16格子や32×32×32格子などのCPU計算の場合より大きなリーフとし、メモリ・アクセスの局所的な連続性を確保する。

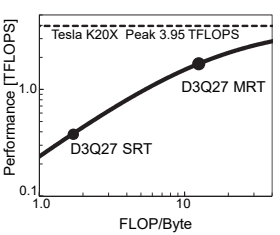


C++ Template によるGPU Kernel自動生成

D3Q27格子ボルツマン法の適合細分化格子では、格子点がリーフ境界のどこにあるかで26通りあり、それぞれに隣接参照のパターンが27通りあるため26×27=702通りの条件分岐がある。条件分岐の無い実装をするため、全隣接パターンのGPUカーネル関数を記述し、複数カーネルの同時実行による高速化を図る。プログラムの生産性を維持しながら記述するため、C++テンプレートを用いて702通りのGPUカーネル関数を自動生成する。

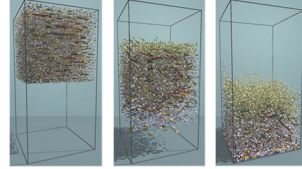


MRT時間積分の演算密度上昇とRooflineモデル



様々なサイズの直方体の沈降

大きさやアスペクト比を変えた5種類の直方体が沈降する計算を行った。格子ボルツマン法による流体計算には512×512×1,024格子点を用い、10,300個の直方体の沈降を調べた。計算にはGPU (NVIDIA Tesla K80)を4台(8 MPI並列)用い、大きい固体粒子が先行して沈降し、小さい固体粒子は沈降速度が遅いため上方に分離する結果を得た。



格子ボルツマン法の時間積分をSRTからMRTに変更すると計算量が増えるため一般的により長い計算時間が必要になるが、演算密度 FLOP/Byte が上昇するため、GPUなどのBF値が低いマシンでは実行性能が上がり相殺される効果がある。図はNVIDIA Tesla K20X で単精度計算をする場合のrooflineモデルによる実行性能の上限がMRTで計算することにより大きく向上することを示している。安定性が増すために Δt も大きく取ることができ、Time-to-Solutionの短縮につながる。

舞い落ちるイチョウの葉の流体構造連成

TSUBAME2.5の128GPUで1,024×1,024×2,048格子点を用い、512枚のイチョウの葉が舞い落ちる過程を計算した。



流動層の大規模直接計算

128GPUによる96×2,560×5,120格子と562,500個の球形粒子を用いた流動層解析を実行した。床面からの流入により粒子が混合されている。右下図のように、粒子周りの流れを捉えた高精度な計算を実現した。

