

粒子法シミュレーションの大規模化と高速化

越塚 誠一

東京大学大学院工学系研究科

1. はじめに

ここでの粒子法は、流体や固体といった連続体に関する力学を粒子の運動に離散化してシミュレーションをおこなう方法を意味しています。具体的には、MPS(Moving Particle Semi-implicit)法やSPH(Smoothed Particle Hydrodynamics)法です。また、粉体を粒子の集まりとして計算するDEM(Discrete Element Method)も加えたいと思います。分子動力学も分子を粒子として計算する方法ですので、プログラミング上は共通するところも多いはずですが、筆者の専門外ですので、ここでは除外することにします。

以下、粒子法シミュレーションのプログラミングに関する特徴、計算例、大規模計算と高速計算、計算結果の信頼性について、最新の研究成果や動向を含めて述べたいと思います。

2. 粒子法と格子法はプログラミングする上で何が違うのか

従来の格子法(有限差分法や有限要素法)でも粒子法でも連続的な空間を離散化することは同じです。図1のように、格子法では空間を格子状に分割して節点に速度などの変数を配置します。一方、粒子法は空間の中に粒子を配置し、その粒子が速度などの変数を持ちます。プログラムではどちらの場合も変数を配列として確保し、その添字が節点番号あるいは粒子番号を意味します。プログラミング言語の構造体の概念を用いれば、各節点あるいは各粒子の持つ様々な変数を1つの構造体の中に記述することもできます。いずれにしても、変数を配列として確保するところは格子法と粒子法で変わりません。

格子法と粒子法の違いは以下になります。(1) 格子法では通常は格子点の座標は固定されていますが、粒子法では粒子の座標は変化します。(2) 格子法では格子点上に配置された変数の関係式を格子線が陽に表していますが、粒子法では近傍の粒子(影響半径 r_e より近い粒子)を時間ステップ毎に探索してこれらとの間に一時的な関係式を作ります。

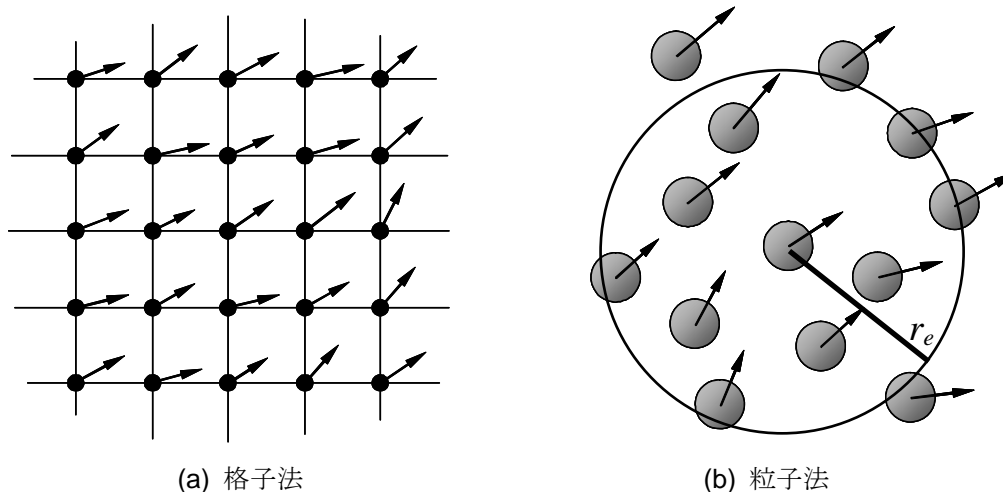


図1 格子法と粒子法

特徴(1)に関しては、粒子の座標を変数として扱う必要があるため計算容量が増加し、その変数の更新のために計算量が増加しますが、これらは相対的に無視できます。特徴(2)に関しては、近傍粒子の探索に粒子間距離を用いますが、この計算には全粒子数を N とすると、 $O(N^2)$ の計算量になります。これは連立一次方程式の解法よりも N の指数が大きいいため、大規模計算では計算時間に対して支配的になります。こうした粒子法の特徴に対する計算アルゴリズムの工夫については後で述べます。

次に、連続体力学の支配方程式は偏微分方程式で記述されますが、これをどのように離散化するかという問題があります。有限差分法では格子幅が差分の基準になりますし、有限要素法では要素の中で積分をおこなって離散式を導出します。こうした格子や要素が無く、単に粒子があるだけで偏微分方程式をどうやって離散化するかが粒子法の本質的なところで

す。MPS 法では粒子間の差分に基づいた粒子間相互作用モデルを用います[1]。差分は、2つの粒子があればその間で作ることができます。ある粒子位置の勾配演算子の粒子間相互作用モデルは、まずその粒子と近傍の粒子との間に差分を作り、その差分の重み平均にします(図2)。同様に発散演算子やラプラシアンに対しても粒子間相互作用モデルを用意します。そして、これらを支配方程式の空間微分に適用すれば、粒子が互いに相互作用しながら運動する式が得られます。これが支配方程式の離散式になります。

SPH 法ではカーネル関数に粒子の持つ変数を掛けて重ね合わせます(図3)[2]。こうすると、格子が無くても変数の空間分布を作ることができます。この変数の空間分布を微分すれば空間微分の離散式が得られます。

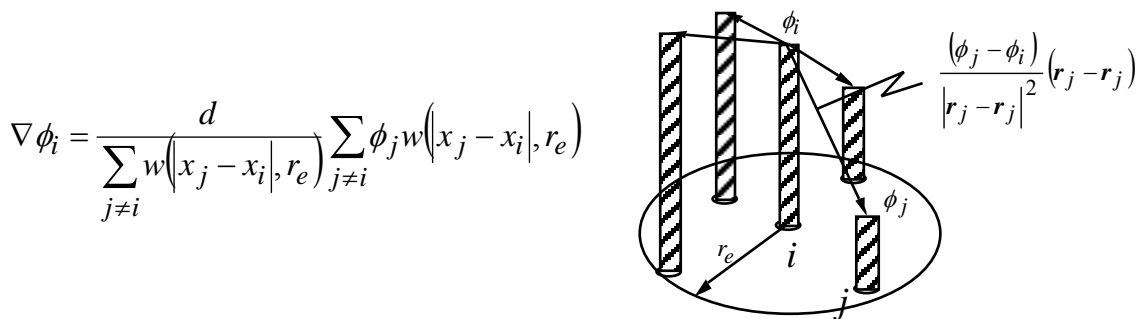


図2 MPS 法における勾配演算子の粒子間相互作用モデル

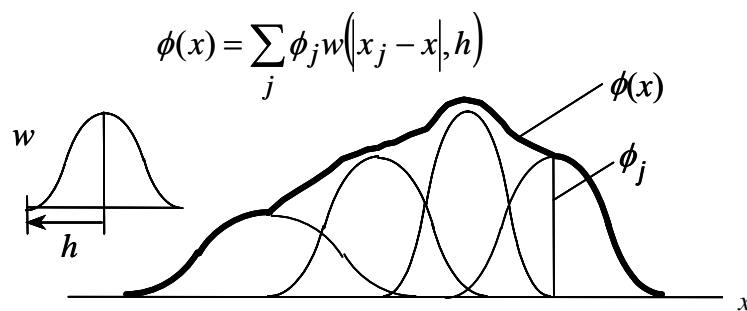


図3 SPH 法における空間分布の構築

MPS法で用いる重み関数 w と、SPH法で用いるカーネル関数 w は、粒子間の離散式を作るうえでほぼ同じ働きがあります。例えばMPS法では多くの場合、以下の重み関数を使います。

$$w(r) = \begin{cases} \frac{r_e}{r} - 1 & 0 \leq r < r_e \\ 0 & r_e \leq r \end{cases} \quad (1)$$

ここで r は粒子間距離、 r_e は影響半径です。 $r < r_e$ であれば w はゼロでない正の値になり、 $r \leq r_e$ ではゼロになります。これによって粒子間相互作用を近傍粒子との間に限定できます。ある粒子に対して、それ以外の全ての粒子との間で相互作用を計算し、重み関数がゼロになることによって遠方の粒子との相互作用を自動的にゼロにできます。ただし、これは $O(N^2)$ の計算量になってしまいます。そこで、近傍粒子を探索し、 $w=0$ とならない近傍粒子とのみ相互作用の計算をすることで、計算量を $O(N^1)$ に落とします。式(1)では $r=0$ で $w \rightarrow \infty$ になりますが、MPS法では同じ粒子との間で粒子間相互作用を計算することは無いので、問題ありません。

MPS法もSPH法も、このように格子を用いることなく微分方程式を離散化する一般的な方法になっています。そのため、流体力学だけでなく固体力学にも適用することができます。粒子法は格子を用いないので、流体力学では自由表面流や気液二相流の液面や界面が大きく変形するような問題、固体力学では大変形や破壊の問題、さらには流体-構造連成問題に適しています。

3. 粒子法の計算例

粒子法がどのような問題に適用されてきたかを紹介します。全体を2種類（エンジニアリングと物理ベースコンピュータグラフィックス）に分けます。それぞれ書籍にまとめられています[3, 4]。

3.1 エンジニアリング

粒子法による非圧縮性流れのシミュレーションでは、自由表面を有する流れを扱うものが最も多いでしょう。その中で基本的なのがダム崩壊問題です。MPS法の最初の論文でもこの問題が解かれています[1]。図4にあるように、初期に水槽の一方の垂直壁に接して水柱を配置し、これが重力で崩れる様子を計算するものです。この問題では、下向きにかかる重力が流体の非圧縮条件を通じて横方向の力に変換され、初期の水柱が広がっていきます。底面上の水の先端の動きは実験と計算で定量的に比較されます(図5)。やがて、もう一方の垂直壁に衝突して水が跳ね上がり、時間の経過とともにその跳ね上がった水が落下します。こうしたふるまいは実験と計算でよく一致しています。

MPS法は自由表面流れに次いで混相流に関する計算例が多くあります。混相流では界面があり、場合によってはそれが大きく変形するので、自由表面流れと同様に粒子法に適した問題です。計算の1例を図6に示します[5]。加熱された底面の上で大気圧の水が沸騰する過程を詳細に計算したものです。加熱壁に配置された初期の気泡が加熱によって成長し、やがて浮力により底面から離脱して上昇します。この問題では壁から離れた位置の水はサブクール（沸点より低い温度）になっているので、加熱壁に接した気泡がある大きさまで成長すると、加熱壁付近での蒸発量と壁から離れた位置での凝縮量がバランスして成長が止まります。また、気泡が加熱壁を離脱して上昇する時には、凝縮によって気泡の体積が減少します。この計算に用いられている粒子法はMPS-MAFL(Meshless Advection Flow-directional Local grid)法で、粒子の

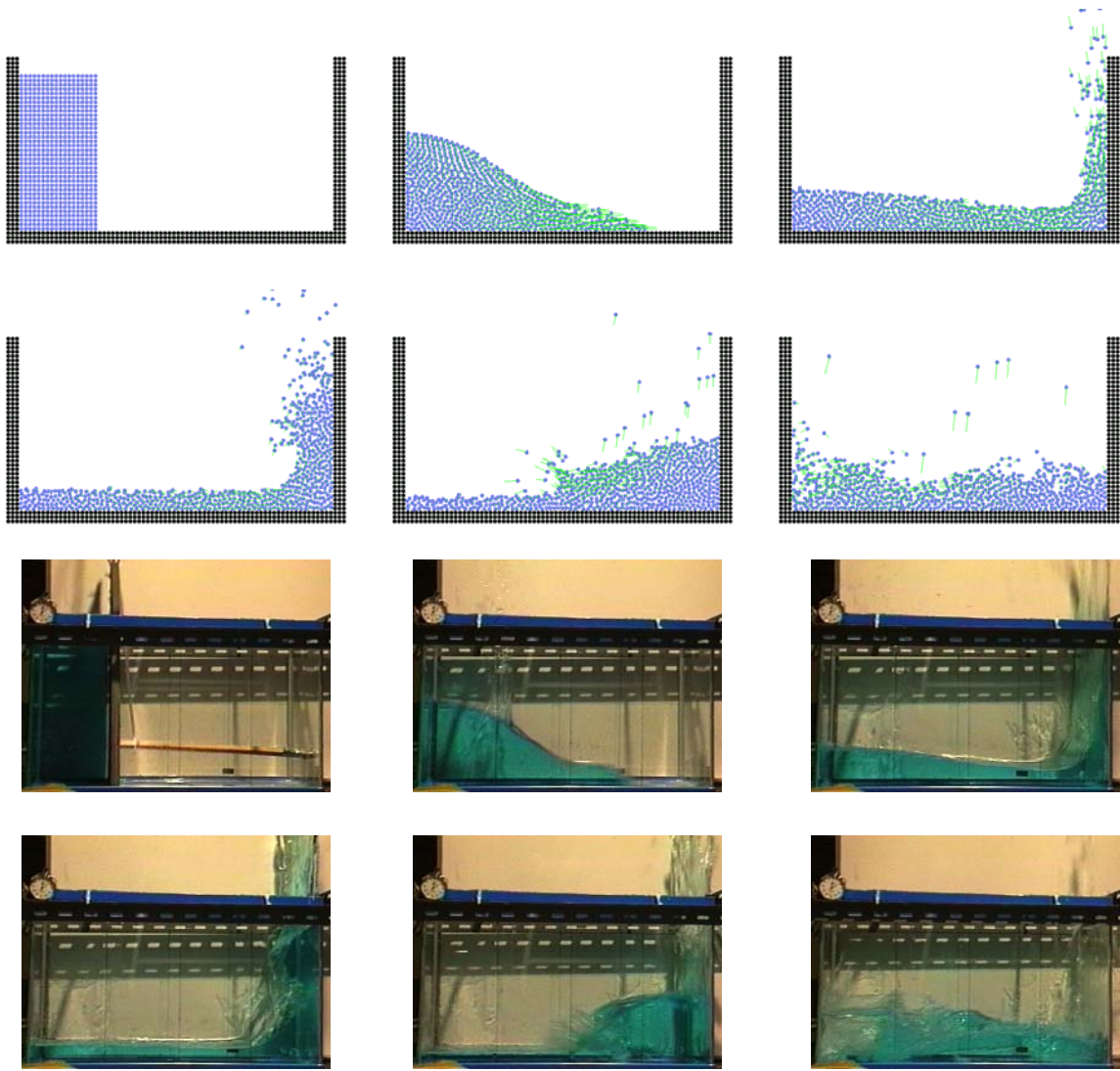


図4 ダム崩壊問題のMPS法による計算と実験

位置を流れとは関係なく任意に決めることができます。これによって、加熱壁と気泡の界面の近傍に小さな粒子を配置し、局部的に空間解像度を上げることができます。もし粒子が流体と一っしょに動いてしまうと、最初に小さな粒子を加熱壁近傍に配置したとしても、時間の経過とともに移動して壁から離れてしまいます。粒子位置を流れとは関係なく決められることで、空間解像度を上げたい場所だけに小さな粒子を配置することができます。また、気泡の成長や上昇とともに小さな粒子の数を増やしており、計算中に総粒子数を変えることも自由にできます。定量的な比較では、気泡半径の時間変化や伝熱量が実験によく一致しています。

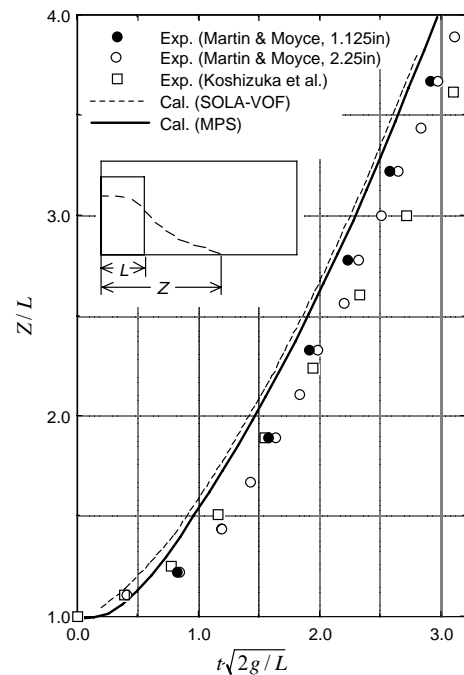


図5 ダム崩壊における底面上の水の先端位置の変化

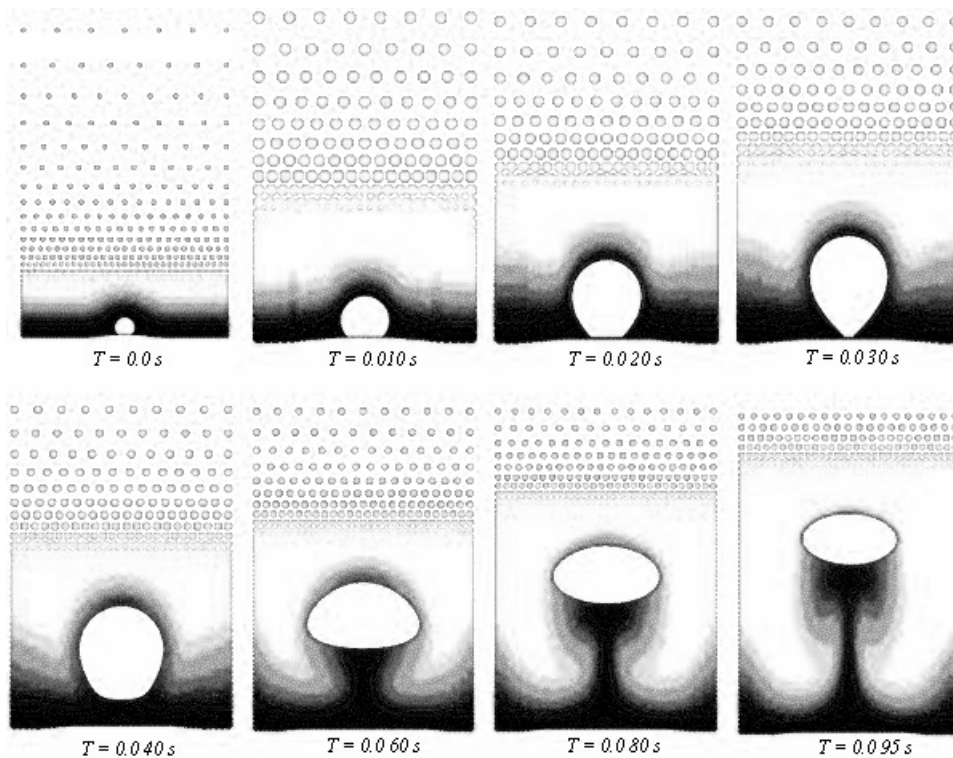
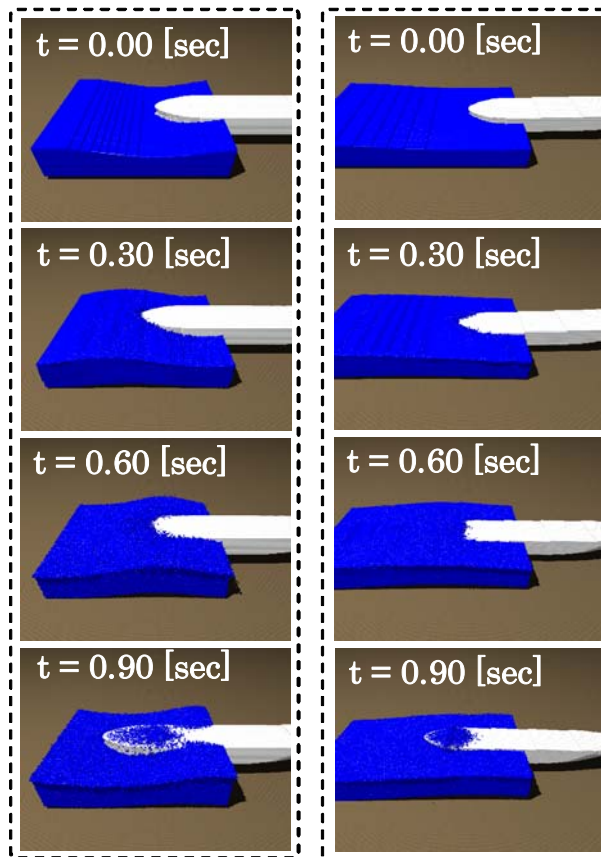


図6 サブクール核沸騰における単一気泡の成長と離脱のシミュレーション（濃淡は温度）

次に、船舶と波浪との相互作用を計算した例を示します。波浪中を航行する船舶では、波が甲板にまで打ち込む現象が起きます。これをMPS法で解析しました[6]。計算結果を図7に示します。この例では模型船を用いた実験が行われていますが、実験で用いた大きな水槽の全体を計算領域に取ることは難しく、ここでは粒子数を節約するために計算領域をできる限り小さくしています。模型船の前方に約1周期分の線形波を作り、側面や底面の壁は線形波の速度分布を用いて動かして、なるべく壁境界の影響が出ないようにしています。模型船の運動は実験では波との相互作用によって生じるのですが、この計算では強制運動として実験で測定されたものを与えています。図7には2例あり、それぞれ波長船長比（＝波長／船長）が0.7と1.5の条件になっています。波長船長比が0.7の場合は船体運動が小さく、波頭に



(a) 波長船長比 0.7 (b) 波長船長比 1.5

図7 船舶甲板への打ち込み水挙動の計算結果

船舶の先端が突入するようにして打ち込みが生じます。波長船長比が 1.5 の場合は船体運動が大きくなり、水面をすくい上げるようにして打ち込みが生じます。こうしたふるまいは実験と良く一致しています。

マイクロサイズの流路を用いた装置は μ TAS (micro Total Analysis System) と呼ばれています。例えば、水相と有機相を単純な Y 字型のマイクロ流路によって合流させるだけで、ほぼ均一なマイクロ液滴を生成することができます。しかしながら、この問題は界面の大変形を伴う液-液の二相流で、しかも表面張力が相対的に大きいので計算が難しくなります。そこで MPS 法の適用を試みました。これについても実験がおこなわれた条件で計算しています。計算結果の 1 例が図 8 です [7]。幅 $100\mu\text{m}$ の流路に水とブチルアセートを流入させています。計算条件は、水相の流量 $Q_{\text{water}}=8.0\mu\text{l}/\text{min}$ 、流入流量比 $R=Q_{\text{org}}/Q_{\text{water}}=1.0$ となっています。合流部で有機相が分裂してマイクロ液滴が生成されています。水相の流量と流入流量比を様々に変えて計算をおこない、液滴サイズに関して計算と実験を比較したところ、定量的に良い一致が得られました。

粒子法を用いたシミュレーションにはこの他にも多くあります。特に、土木工学の分野では後藤仁志先生が精力的に研究を進めています [8]。原子力工学においても高速炉の過酷事故解析のために、MPS 法を用いた多相多成分流れの解析コードの開発が進められています [9]。構造力学の分野では酒井譲先生が SPH 法を使って様々な問題を解いています [10]。MPS 法による例もあります [11]。

3.2 物理ベースコンピュータグラフィックス

物理的に正しいシミュレーションをおこない、その結果にもとづいたコンピュータグラフィックスにより映像を制作することを、物理ベースコンピュータグラフィックス (物理ベース CG) と呼びます。流体解析、固体解析、剛体解析、布の解析、相変化などの様々な物理現象の粒子法シミュレーション手法がこれまでに研究されています。流体解析や固体解析では MPS 法や SPH 法といった方法が使われるのに対し、剛体解析では DEM (Discrete Element Method) が使われます。粒子法では界面の複雑な動きのような激しい現象を扱うことが容易なため、映像制作のためのシミュレーション手法に適しています。また、粒子法では粒子が新たに現れたり消えたりしなければ質量保存が成立します。多少の計算誤差があっても質量保存が崩れないという性質は高速計算の上で利点です。さらに様々な物理現象を粒子間相互作用という統一的な考え方で計算できるため、マルチフィジックスシミュレーションが比較的容易にできます。そのため、物理ベース CG において粒子法は広く用いられるようになってきています。

例えば、粒子法によってコップに水を注ぐシミュレーション

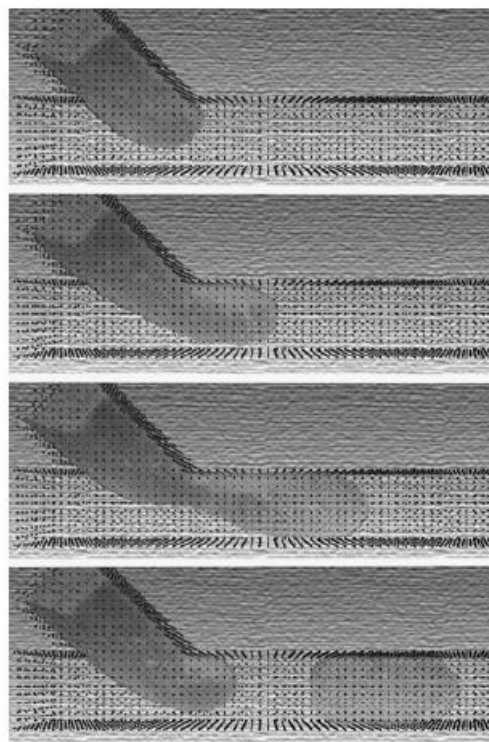


図 8 マイクロ液滴生成の計算結果

をおこない、その結果にもとづいて作成した画像が図9です[12]。この例では気泡の計算モデルが組み込まれていて、これによって画像が本物らしくなっています。図10は剛体の物理ベースCGです[13]。1個の剛体を複数の粒子で形状を模擬し、剛体間の衝突や摩擦は粒子にもとづいて計算します。従来のポリゴンにもとづいた方法と比較して、剛体間の衝突や相互作用をきわめて効率的に計算することができます。描画はポリゴンにもどしておこないます。同様に、弾性体を粒子群に離散化してMPS法を適用した物理ベースCGを図11に示します[14]。この場合、変形に対応して弾性体を構成する粒子間の相対位置が変化します。この変形の情報ポリゴンの頂点の座標に反映させて描画しています。連成解析の例として、流体と剛体を組み合わせた例を図12に示します[15]。積み上げたグラスは剛体



図9 流体の物理ベースCG

として計算され、そこに赤い色の流体を注いでいます。新たなグラスを投げつけると積み上げたグラスが崩れますが、これは剛体としてのシミュレーションをおこなっているため、得られる映像は本物らしくなります。剛体も流体も粒子によって計算されているため連成は容易です。

こうした物理ベースCGの技術は、映画、テレビ番組、コンピュータゲームといった映像産業に既に広く用いられています。また、映像産業だけでなく、災害対策、仮想外科手術、操船シミュレータ、バーチャルファッションショーなど、様々な産業応用も考えられています。

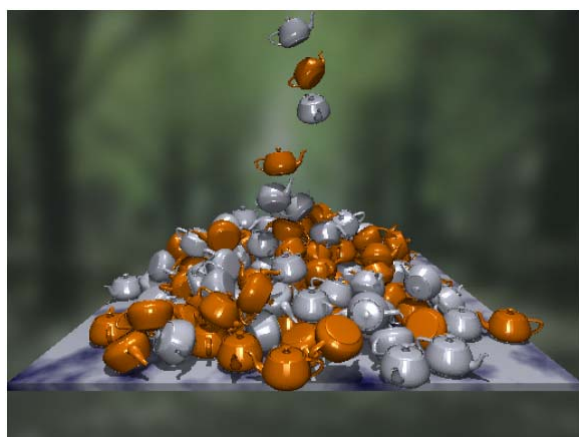


図10 剛体の物理ベースCG

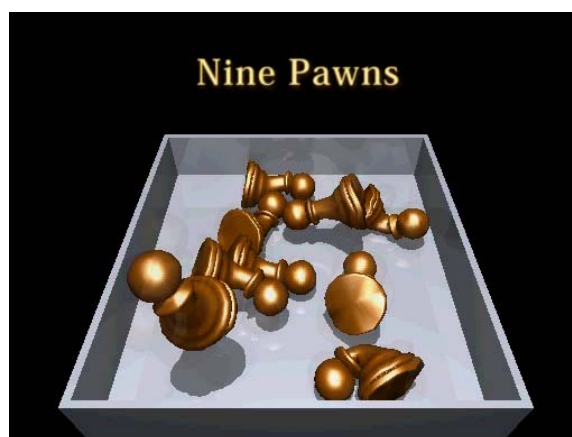


図11 弾性体の物理ベースCG

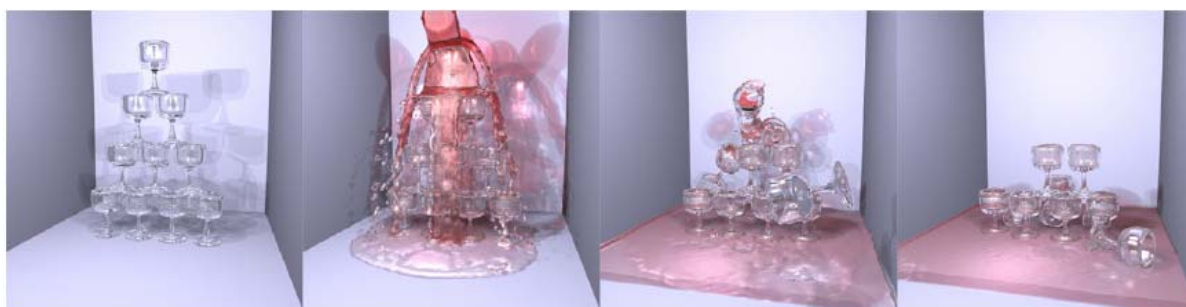


図12 流体と剛体の連成の物理ベースCG

4. 大規模化と高速化

4.1 粒子法特有の問題に対するアルゴリズムの工夫

2章で粒子法のプログラミング上の特徴をまとめました。ここではその特徴に対する並列計算手法などのアルゴリズムの工夫を述べます。図1に戻って、大規模高速計算の観点から粒子法の特徴を整理してみます。

まず第1に粒子の座標が変数で、時々刻々変化するということがありました。この計算は、粒子 i の古い時刻の座標と速度をそれぞれ $r[1][i]$, $u[1][i]$, 新しい時刻の座標と速度をそれぞれ $rn[1][i]$, $un[1][i]$ とすると (1 は空間次元), C 言語では以下のように書けます。

$$rn[1][i]=r[1][i]+dt*un[1][i]; \quad (2)$$

すなわち、古い時刻の粒子座標 $r[1][i]$ に、時間刻み幅 dt と新しい時刻の粒子速度 $un[1][i]$ を掛けたものを足します。粒子座標の更新の計算は、どのような物理現象を解く場合でも基本的には共通です。

新しい時刻の速度 $un[1][i]$ は運動量保存則から計算します。

$$un[1][i]=u[1][i]+dt*f(r, u); \quad (3)$$

運動量保存則における右辺の $f(r, u)$ は流体であれば流体の運動量保存則, 弾性体であれば弾性体の運動量保存則によって計算します。つまり、解きたい物理現象によってここは異なります。式(2)と(3)の順番については、式(3)の計算が先で、その結果である $un[1][i]$ を用いて式(2)を計算します。

さて、粒子の座標が時々刻々変化するのを計算するのは具体的には式(2)であり、相対的には無視できる程度の計算量にしかありません。粒子数を N とすると $O(N)$ の計算量であり、大規模になっても相対的な計算量は増えません。

第2に各時間ステップで近傍粒子を探索する必要があります。図1に示すように粒子法では影響半径 r_e 内の近傍粒子を探索します。この計算のために全ての粒子の間の距離を計算し、これが r_e 以下かどうかを判定すると、 $O(N^2)$ の計算量になります。距離の計算は単純で粒子数が少なければ計算量は少ないのですが、連立一次方程式の解法の計算量が $O(N^{1.5})$ 程度ですので、粒子数が多くなると距離の計算に必要な計算量は相対的に増加し、100万粒子ともなるとほとんど全ての計算時間を距離計算に費やしてしまうこととなります。そこで、これを回避するためにMPS法ではこれまで2種類の方法が使われてきました。

1つは、より大きな影響半径による近傍粒子テーブルをもう1つ作成しておくことです。この広範囲の近傍粒子テーブルから影響半径 r_e の本来の近傍粒子テーブルを作成します。一方、広範囲の近傍粒子テーブルは毎時間ステップに更新する必要はありません。これで計算量は $O(N^2)$ から $O(N^{1.5})$ に落ちます[16]。もう1つの方法は空間を格子状に分割して、粒子がどのバケット(格子線で囲まれた領域, セル)に入っているかを登録しておくものです。ある粒子の近傍粒子は、その粒子の入っているバケットとその周囲のバケットに限定して探索すればよく、計算量は $O(N^{1.0})$ まで落ちます[17]。後者の方法は実装が簡単なので、現在のMPS法の大規模計算では広く使われています。

格子法において離散式を構成する格子点の数は、2次元では図1を見ると線につながっている上下左右および自分を合わせて5点になります。一方、粒子法においては影響半径 r_e 内にある粒子数は10個を数えます。このように、粒子法において離散式を構成する粒子数は、格子法において離散式を構成する格子点よりも多くなります。影響半径 r_e を小さくしていけば近傍

の粒子の数は減るのですが、減りすぎると計算が不安定になるという問題が出てきます。安定に計算ができる程度の大きさの影響半径 r_e を取ると、近傍粒子の数は多くなってしまいます。これは計算時間の増大につながります。

粒子法では離散化式が空間の次元によってほとんど変化しません。粒子間距離の計算に次元の数が影響しますが、粒子法の離散式では粒子間距離を使いますので、粒子間距離を求めた後の計算は空間次元の影響をほとんど受けません。従って、粒子法では3次元計算コードの作成が容易です。

PCクラスタで並列計算をする場合、粒子番号によって分割して異なるCPUに担当させることが考えられます。粒子番号による分割が空間による分割と対応していれば、近傍粒子同士が同じCPUの担当になっている場合が多くなり、CPU間の通信が少なくて済みます。しかしながら、粒子法では粒子が時々刻々移動するため、粒子群がお互いに混ざり合うことになります。そうすると、他のCPUが扱う粒子が近傍粒子となる場合が増え、CPU間の通信が増大します。これは並列計算の効率を悪化させます。そこで、ある程度計算が進んで粒子群が混ざってきたところで、粒子番号を空間分割に合わせて再設定します[18]。こうすると、CPU間の通信量をあまり増大させることなく、粒子法による並列計算を続けることが可能になります。粒子番号の再設定において、近傍粒子探索に用いたバケットを利用することもできます。

また、近傍粒子探索で用いたバケットによって領域分割して、異なるプロセッサに割り当てることも考えられます。この場合、各プロセッサの担当する粒子数が一定ではなくなるという問題があります。複数のGPUを用いた並列計算ではこのバケットを用いた分割の方法が用いられています[19, 20]。

4.2 GPUの利用

GPU(Graphics Processing Unit)は描画を高速におこなうための専用のプロセッサですが、近年プログラムによって動作させることが可能になり、科学技術計算に利用する試みがおこなわれています。プラズマの差分法による計算(ポアソン方程式を解くステップも含む)においてGPUとCPUを比較し20倍の計算速度が得られ、移流項部分だけでは85倍の計算速度が得られたとの報告があります[21]。粒子法のシミュレーションやこれに関連する前処理におけるGPUの利用に関する研究は、原田隆宏氏が多数おこなっています[4, 19, 20, 22]。最近では大規模体系でCPUの約100倍の性能が出たとの報告があります[20]。複数のGPUを用いて粒子法の並列計算をおこない、4GPUで単一GPUの約3倍の性能が得られたと報告されています[19, 20]。この研究ではGPUの並列計算に適した近傍粒子探索アルゴリズムおよび粒子番号の再設定のアルゴリズムが提案されています。GPUは極めて安価ですので、今後ますます科学技術計算に利用されていくのではないかと思います。

4.3 大規模化と高速化のニーズ

シミュレーションの大規模化と高速化には現在においても大きなニーズがあります。シミュレーションの目的は研究、設計、トラブル解明、評価など様々ですが、ここでは実用的なシミュレーション、すなわち産業において使われることを考えてニーズを考察してみます。

まず産業で考えられるのが、製造業における設計や評価への利用です。この場合は計算結果の信頼性が重要です。計算結果の信頼性の中身については5章で議論したいと思います。信頼

性の高い計算をするためには、十分な空間解像度、十分な時間解像度、十分な反復計算の収束が必要であるだけでなく、複雑な形状の製品の模擬性を良くしたり、様々な物理現象の複合的な効果を考慮する必要があります。どの要求に対しても計算負荷の増大につながります。しかも予算と時間の制約の中で実行しなければなりません。現在の計算機的能力では、実際の製品を非常に単純化しなければ設計のための計算ができないという場合がまだ多いでしょう。そもそもシミュレーションでは精度の高い設計ができないという場合もしばしばあると思います。計算の大規模化と高速化が進めば、設計や評価においてシミュレーションが使える対象が増えますし、既にシミュレーションが使われている場合には設計コストの低減、時間短縮、精度向上による設計マージンの低減が進められます。最近ではシミュレーションによる最適化やロバスト設計が試みられるようになっていますが、この場合には入力パラメータを変化させて多数のシミュレーションを実行する必要があり、ここでも計算速度の向上が求められています。

物理ベース CG においても、大規模化と高速化には強いニーズがあります。映画でコンピュータグラフィックスが多用できるのは、計算機のパフォーマンスの向上によってシミュレーションと CG の大規模化と高速化が可能になったためでしょう。映画のための映像制作はオフラインでよいのですが、運転訓練シミュレータやゲームはリアルタイムである必要があります。ここで要求される高速化は非常に厳しいものです。これまでは科学技術計算レベルのシミュレーションはリアルタイムでは不可能でした。しかし最近では、特に GPU を用いたシミュレーションでは、これが現実的になってきています。

4.4 HPC の展望

HPC (High Performance Computing) については、ハードウェア開発および計算機の設置のどちらについても日本の占める位置が相対的に低下しています。例えば、スーパーコンピュータの TOP500 のリストが半年毎に公開されていますが、最新の 2008 年 6 月のリストでは、プロセッサのベンダーは IBM 68, Intel 375, AMD 55, NEC 1, Cray 1, その他 0 となっており、マイクロプロセッサの寡占化が進んでいることがわかります[23]。TOP1 は米国ロスアラモス研究所のマシンで、プロセッサは IBM の PowerXCell 8i, ピーク性能は 1.37PFlops となっています。日本設置のマシンに限定した TOP4 (世界では 16, 20, 24, 34 位) では、いずれも AMD のマイクロプロセッサが使われています。

4.2 で述べたように、最近 GPU を用いた科学技術計算の高速化が注目されています。単精度計算ですが、わずか 2 万円程度のグラフィックカード (例えば、nVIDIA 社の GeForce8800GTS (G92) 512MB, コア数 128 個) によって CPU よりも 2 桁速い計算速度が得られています。GPU の技術の進歩は速く、既に次の世代の GPU (例えば、nVIDIA 社の GeForce GTX280, コア数 240 個) が発売されており、その性能は以前のもの 2 倍近くになっています。こうした GPU はプログラミング可能で、nVIDIA 社は CUDA という言語を提供しています。一方、Intel 社もこれまでのマイクロプロセッサの延長線上に新しい GPU (Larrabee) を独自に開発しています[24]。

こうしたグローバルな状況において、今後の HPC の動向を予想するには、研究のための科学技術計算における HPC だけを見ていたのでは不十分で、産業利用のための安価で使いやすい HPC の市場や、映像制作やゲームにおける研究開発の動向まで広く見る必要があると思います。まず第 1 に、現状では TOP500 のマシンのうち 498 台が 3 社のマイクロプロセッサに占められており、汎用マイクロプロセッサの開発動向に HPC は従属していると言わざるを得ません。次に、

これに対する新しい動きとして GPU が HPC に参入すると予想されます。GPU を用いた高速な科学技術計算は、リアルタイム性を要求されるゲーム分野の物理シミュレーション技術として注目されており、その動向が今後の HPC のありかたに大きな影響を与えることも考えられます。

5. シミュレーションの信頼性

本稿ではシミュレーションの Verification と Validation (V&V) について紹介します。V&V とはシミュレーション結果の信頼性を定量化する方法で、数値流体力学に対しては米国航空宇宙学会 (AIAA) の 1998 年のガイドライン[25]が、計算固体力学に対しては米国機械学会 (ASME) の 2006 年のガイドライン[26]があります。Verification と Validation ですが、どちらも日本語に訳すと「検証」になります。英語の日常会話でも Verification と Validation は類義語で区別がないそうです。しかし、ガイドラインでははっきりと区別します。

工学の目的でシミュレーションをおこなう手順を考えます (図 1 3)。まず、実際に解きたい現実世界の問題に対して、本質的な物理現象だけを抽出した概念モデルを作ります。この概念モデルを元に計算モデル (計算コード) を作成し、計算結果を得ます。AIAA のガイドラインでは、Verification とは概念モデルと計算結果との一致を定量化するものです。具体的作業としては、概念モデルを記述する微分方程式の解析解と、その微分方程式を離散化して得られた計算モデルの数値解との比較です。両者の違いの原因には、空間の離散化誤差、時間の離散化誤差、反復計算の収束誤差、プログラムミスや入力ミスが考えられます。一方、Validation では現実世界の問題と計算結果との一致を定量化するものです。具体的作業としては、現実世界の問題を反映する実験をおこなって、その実験結果と計算結果の違いを比較します。現実世界の問題は複合問題が多いので、それらを階層構造でとらえて、様々なレベルでの Validation の積み重ねが必要になります。ただし、予算などの制約条件によって V&V は影響を受けるというような記述もあって、現実的なガイドラインになっています。

ASME のガイドラインでも Verification と Validation の定義はほぼ同じです。ただし、Verification を Code Verification と Calculation Verification の 2 つに分けています。また、Validation において使用目的に限定されることを強調しており ISO9000 の考え方が入って

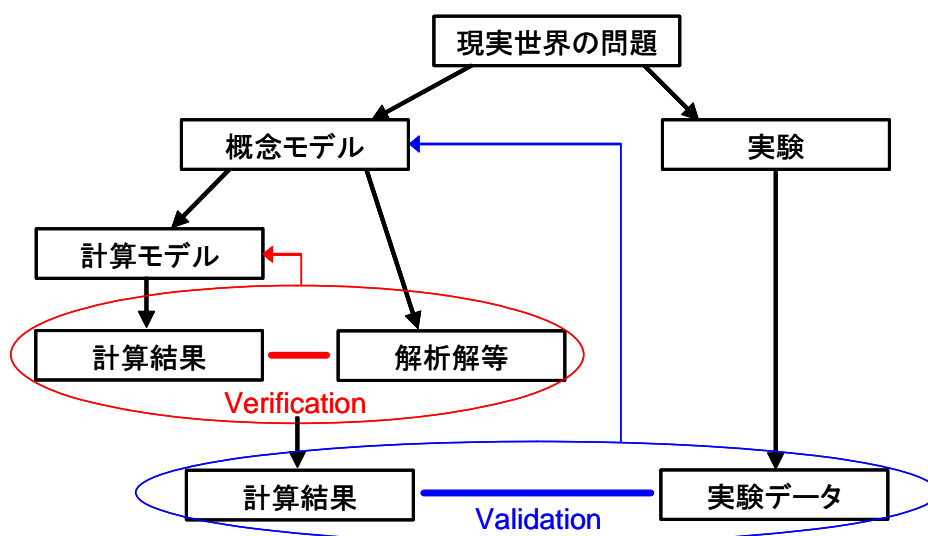


図 1 3 Verification と Validation (V&V) の手順

います。ISO9000 はより一般的なソフトウェア開発における品質保証の技術基準であり、そうした観点から Verification と Validation についての別の定義が示されており、Validation については使用目的に限定されることが書かれています。さらに、ASME のガイドラインでは概念モデルの構築に重要度分類表(PIRT)の利用を推奨しておりますが、PIRT は原子力プラントの統計的な安全解析で用いられている考え方で、それを一般的な V&V に取り入れているというものです。

こうした米国のガイドラインに沿って Verification と Validation をおこなっていくことが、シミュレーション結果の信頼性を定量化するには、現状では最良と思われれます。ただ昨今では、ガイドラインを陽に用いていなくても、これと類似の手順に沿って研究開発がおこなわれている場合が増えていると思います。ガイドラインはそれらを整理して1つの手順にまとめたものと見ることもできます。

ちなみに、これらのガイドラインはシミュレーションや科学技術の本質に関する哲学的な問題にも触れています。例えば AIAA のガイドラインには、prediction は未知の現象の予測であって V&V によって prediction の精度を直接語るができない、と書かれています。この問題は、筆者が 2007 年 7 月に San Francisco で ASME のガイドラインに関する特別セッションに出席したときにも議論になっていました。V&V で語れる精度が postdiction に関するものだけであるとしたら、シミュレーションの有用性はいったい何なのかということになりますね。極論すれば、シミュレーションとは既に分かっていることをコンピュータで再現しているだけだ、ということです。AIAA のガイドラインではこの問題は先送りされていると思います。

一方、ASME のガイドラインには、「科学理論や予測の真実性は演繹的な論理で証明されるものではない。それらは証拠(evidence)の積み重ねによって正しさが試される。証拠は V&V によって系統的に組織化される」という記述があります。これは上記の問題に対する科学技術の態度を適切に言い表していると思います。さらに議論の対象を予測(prediction)だけでなく科学理論(scientific theory)にまで広げていることから、ガイドラインの作成者が科学哲学を意識していると推察できます。

さて、シミュレーションの信頼性は V&V だけで完結するものではありません。プログラムミスや入力ミスは、品質保証の問題と技術者の技量の問題を含んでいます。品質保証の問題には例えば ISO9000 の方法論を適用すればいいですし、技術者の技量に対しては例えば日本機械学会による計算力学技術者認定試験の制度があります。シミュレーションが学術や産業において広く使われるようになってくると、その信頼性が社会にとって極めて重要であることは、少し前の耐震計算に関わる偽装事件を挙げるまでもなく、明らかです。シミュレーションの技術者には結果の信頼性に対して責任があるというような記述が ASME のガイドラインにも書かれています。これは技術者倫理の問題とすることができます。

6. おわりに

本稿では粒子法シミュレーションの特徴や計算例の紹介をおこないました。また、大規模計算と高速計算に関しては、粒子法に関する並列計算手法などのアルゴリズムの工夫と GPU を用いた高速計算を解説した後、大規模化と高速化のニーズと HPC に関する私見を述べました。最後に、計算精度に関しては筆者が最近気になっている米国における V&V のガイドラインを紹介しました。内容が様々なテーマに分散しておりますが、部分的にでも読者の興味に合うところ

があれば幸いです。

参 考 文 献

- [1] Koshizuka, S. and Oka, Y., "Moving-Particle Semi-implicit Method for Fragmentation of Incompressible Fluid," *Nucl. Sci. Eng.*, **123**, 421-434 (1996)
- [2] Lucy, L. B., "A Numerical Approach to the Testing of the Fission Hypothesis", *Astron. J.*, **82**, 1013-1024 (1977)
- [3] 越塚誠一, "粒子法," 丸善 (2005)
- [4] 越塚誠一, 原田隆宏, 田中正幸, 近藤雅裕, "粒子法シミュレーション —物理ベースCG入門—," 培風館 (2008)
- [5] Yoon, H. Y., Koshizuka, S. and Oka, Y., "Direct Calculation of Bubble Growth, Departure and Rise in Nucleate Boiling," *Int. J. Multiphase Flow*, **27**, 277-298 (2001)
- [6] 柴田和也, 越塚誠一, 谷澤克治, "船体の強制運動を伴う海水打ち込みの粒子法による3次元解析," *日本船舶海洋工学講演論文集第1号*, 福岡, November 24-25 (2005) p. 59-60
- [7] 原田隆宏, 鈴木幸人, 越塚誠一, 荒川貴博, 庄子習一, "MPS法を用いたマイクロ混相流3次元解析," *日本機械学会論文集(B)*, **73**, 437-444 (2007)
- [8] 後藤仁志, "数値流砂水理学 —粒子法による混相流と粒状体の計算力学—," 森北出版 (2004)
- [9] Koshizuka, S., Liu, J., Shirakawa, N., Uehara, Y., Naitoh, M. and Yamamoto, Y. "R & D of the Next Generation Safety Analysis Methods for Fast Reactors with New Computational Science and Technology (1) Introduction of the Project and Development of Structural Mechanics Module," *Proc. 16th Int. Conf. on Nuclear Engineering (ICONE16)*, Orlando, May 11-15, 2008, ICONE16-48499
- [10] 酒井讓, "粒子法による構造解析の動向とトピックス," *粒子法研究会 (第十回) 報告書, UTNL-R-0454*, 東京, March 8, 2006, p.1-14
- [11] 宋武燮, 越塚誠一, 岡芳明, "MPS法による弾性構造体の動的解析," *日本機械学会論文集(A)*, **71**, 16-22 (2005)
- [12] 須藤健太郎, 越塚誠一, "粒子法による泡の運動シミュレーション," *第19回計算力学講演会講演論文集*, 名古屋, November 3-5, 2006, p.703-704
- [13] Tanaka, M., Sakai, M. and Koshizuka, S., "Rigid Body Simulation Using a Particle Method," *DVD publication at ACM SIGGRAPH posters*, Boston, July 30 - August 3, 2006, 132
- [14] Kondo, M., Tanaka, M., Harada, T. and Koshizuka, S., "Elastic Objects for Computer Graphic Field Using MPS Method," *DVD publication at ACM SIGGRAPH posters*, San Diego, August 5 - 9, 2007, sap-179
- [15] Harada, T., Tanaka, M., Koshizuka S. and Kawaguchi, Y., "Real-time Particle-Based Simulation on GPUs," *DVD publication at ACM SIGGRAPH posters*, San Diego, August 5 - 9, 2007, sap-151
- [16] Koshizuka, S., Nobe, A. and Oka, Y., "Numerical Analysis of Breaking Waves Using the Moving Particle Semi-implicit Method," *Int. J. Numer. Meth. Fluids*, **26**, 751-769

- (1998)
- [17] 後藤仁志, 林稔, 酒井哲郎, “粒子法による直立護岸前面砕波・越波の数値シミュレーション,” *土木学会論文集*, 第 726 号, 87-98 (2003)
 - [18] 入部綱清, 藤澤智光, 越塚誠一, “粒子法による大規模解析におけるノード間通信の低減,” *Trans. Japan Society for Computational Engineering and Science*, Paper No. 20080020 (2008)
 - [19] 原田隆宏, 政家一誠, 越塚誠一, 河口洋一郎, “複数の GPU を用いた粒子法シミュレーションの並列化,” *第 13 回計算工学講演会論文集*, **13**, 仙台, May 19-21, 2008, p. 289-290
 - [20] 原田隆宏, “GPU を使った並列処理で物理シミュレーションを高速化,” *日経エレクトロニクス*, 2008 年 6 月 2 日号, p. 105-115 (2008)
 - [20] 小川慧, 青木尊之, “GPU を用いた CIP 法によるプラズマ 2 流体不安定性の高速シミュレーション,” *第 13 回計算工学講演会論文集*, **13**, 仙台, May 19-21, 2008, p. 837-840
 - [22] Harada, T., “Real-time Rigid Body Simulation on GPUs,” *GPU Gem 3*, Addison-Wesley (2007)
 - [23] <http://www.top500.org/>
 - [24] Seiler, L., Carmean, D., Sprangle, E., Forsyth T., Abrash, M., Dubey, P., Junkins, S., Lake, A., Sugerman, J., Cavin, R., Espasa, R., Grochowski, E., Juan, T. and Hanarahan, P., “Larrabee: A Many Core x86 Architecture for Visual Computing,” *ACM Trans. Graphics*, **27**, Article 18, 2008
 - [25] American Institute of Aeronautics and Astronautics, “Guide for the Verification and Validation of Computational Fluid Dynamics Simulations,” *G-077-1998* (1998)
 - [26] American Society of Mechanical Engineers, “Guide for Verification and Validation in Computational Solid Mechanics,” *ASME V&V10-2006* (2006)