

推薦論文

SPH ベースリアルタイム火炎シミュレーション

間 淵 聡^{†1} 藤 代 一 成^{†1} 大 野 義 夫^{†2}

本論文では、火炎の粒子ベースシミュレーション法を提案する。本手法は、SPH による流体シミュレーションを火炎に適用したものであり、ナビエ・ストークス方程式の外力として、シンプルな化学反応モデルから発生した熱による浮力項を加える。すなわち、既存の SPH のプログラムと十分な互換性を保持することができ、複雑な物理シミュレーションプログラムの一元化が可能である。また、火炎の周辺に粒子が集中するように効果的にリサンプリングすることで計算コストを抑える。本手法を用いることで、火炎の形成から安定状態への推移、固体からの 1 方向の作用による変形などが滑らかに表現でき、リアルタイムに 3 次元シミュレーションを実行することができる。

SPH-based Real-time Flame Simulation

SATOSHI MABUCHI,^{†1} ISSEI FUJISHIRO^{†1}
and YOSHIO OHNO^{†2}

In this paper, we propose a novel particle-based flame simulation method, which relies on SPH to simulate the dynamics of flame by introducing a buoyancy term as an external force to Navier-Stokes equation. The buoyancy term is directly affected by heat generated from a simple chemical reaction model. Good interchangeability allows us to easily couple our method with existing SPH source codes. For efficient execution, we adaptively re-sample particles around flame. We demonstrate three-dimensional simulations in real-time for typical cases including transition from initial to stable states, and one-way coupling with solid.

^{†1} 慶應義塾大学

Keio University

^{†2} 東北公益文科大学

Tohoku University of Community Service and Science

本論文の内容は 2010 年 9 月のグラフィクスと CAD 研究会にて報告され、同研究会主催により情報処理学会論文誌ジャーナルへの掲載が推薦された論文である。

1. 序 論

CG における物理ベースシミュレーションは固体、液体などのさまざまな現象でリアルタイム性を提供できるようになりつつある。しかしながら、火炎のシミュレーションにおいては十分なリアルタイム性のある手法が確立されていない。本論文では通常の PC を用いて再現可能な、物理的妥当性があり、空間制約のない、火炎の粒子ベースシミュレーション法を提案する。

近年では流体シミュレーションに粒子法の 1 つである SPH (Smoothed Particle Hydrodynamics) を用いることが多くなった。SPH は物理的妥当性があることと高速計算が可能であることから、さまざまな応用研究がなされている。しかし多くは液体を対象としたものであり、気体や火炎への適用はほとんどなされていない。本論文では SPH を火炎に拡張した手法を提示する。

関連研究では爆発火炎のように安定状態を形成しないものが多いが、本研究ではろうそくやバーナのような、日常的に目にする、安定的に形成される比較的小規模な火炎の再現に主眼を置いた。

本論文は著者らの研究報告²¹⁾ をベースとしているが、一部に改良を加えた。具体的には、リサンプリングのアルゴリズムを総当り方式から法線を用いる方式にし、また画面のズームに耐えうるように、レンダリング手法に修正を加えた。

本論文では、次章で関連研究を紹介した後、3 章で関連研究を土台としたアプローチを説明する。4 章で、計算モデルとその離散化手法を述べ、5 章で、計算モデルの高速な実装方法を説明する。最後に、6 章で実装したシミュレーションの結果を、7 章で結論と課題を述べる。

2. 関連研究

CG における火炎を含む流体に関する研究は、格子法と粒子法とに大別されている。液体、気体については特に区別せずに紹介を行うが、性質の大きく異なる爆発火炎については述べない。

2.1 格子法

火炎シミュレーションを初めて格子ベースで導入したのは、Inakage¹²⁾ である。この研究では化学反応を扱っているが、対象はかなり限定的なシーンである。ハードウェアの発展にともなって、流体シミュレーション法を CG に導入したのは Foster ら⁶⁾ である。これは写

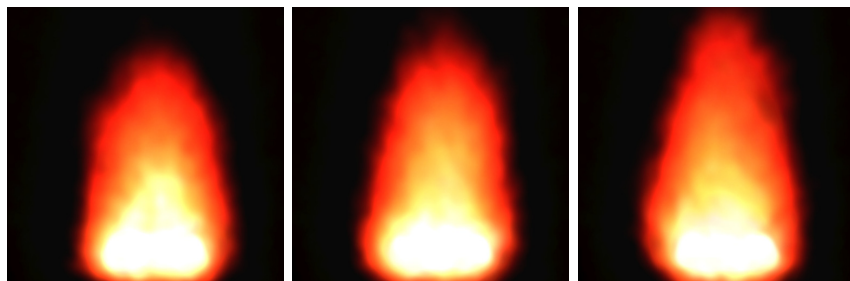


図 1 提案手法による火炎のリアルタイムシミュレーション例
Fig.1 Flame simulation with our method in real-time.

実性に問題があったが、Level Set 法を用いることで十分な写実性を提供できるようになった^{3),5)}。格子法では液体に限らず、さまざまな物理現象がシミュレートされている。特に気体に関しては、煙のシミュレーション⁴⁾、火炎シミュレーション¹⁶⁾などが行われている。さらに火炎のディテールの表現を追求した研究⁹⁾もある。これらは、非圧縮性を満足させるために、大規模連立一次方程式を解く必要があり、リアルタイム性を持たせることは難しかった。近年では、高速化のため、少数の粒子によるシミュレーションを CPU 上で行い、詳細なシミュレーションとレンダリングを複数の GPU 上で行う手法が Horvath ら¹⁰⁾によって提案された。また、固体の燃焼をリアルタイムに扱った研究も報告されている^{13),19)}。

格子法では想定するシミュレーション空間を事前に用意しておく必要がある。また、リアルタイム性を要求される場合には、格子セル数を大きく減らす必要があり、対話的に扱うことのできる空間に強い制限が加わる。

2.2 粒子法

CG における流体アニメーションを最初に発表したのは Reeves¹⁷⁾である。この研究では、炎の生成から消滅までを単純化し、80 万程度の粒子を用いてそれらしく見せることに成功している。その後しばらくは物理法則をあまり考慮しない、アニメーションとしての研究が進められたが、SPH の導入²⁾によってある程度の物理法則が考慮され、Müller ら¹⁴⁾は SPH を拡張してリアルタイム流体シミュレーションを実現した。さらに近年では Harada ら⁸⁾が、GPU を用い、速度を CPU の数十倍に引き上げた。井村ら²⁰⁾は SPH ベースでの火炎レンダリング手法を提案したが、シミュレーションには簡易的なモデルが用いられている。このように近年では SPH の発展が目覚ましいが、液体シミュレーションが主流となっており、気体や火炎への適用例は少ない。

粒子法においては、シミュレーション空間を事前に想定する必要がないため、対話的に扱うことのできる空間への制限がほとんどない。

3. アプローチ

前章で述べたとおり、現在までの火炎シミュレーションは、格子法では空間制約が強く、粒子法では物理的根拠に欠けている。これをふまえ、本論文では、空間制約が少なく、物理的根拠のある、SPH による火炎シミュレーション法を提案する。主な特長は物理的妥当性とリアルタイム性であり、詳細を以下に述べる。

3.1 物理的妥当性

気体は、物理的には、流体の支配方程式（ナビエ・ストークス方程式）に従う。したがって、SPH を物理的シミュレーションのベースとして用いる。提案手法は既存の SPH プログラムに、火炎シミュレーションに必要な部分を追加するだけで実現が可能である。これは同時に、既存の SPH と同等の物理的妥当性を保証することを意味する。

一方、化学的妥当性に関しては、比較的複雑な化学反応を考慮した研究¹¹⁾もあるが、調整パラメータの増加につながるため、単純化したモデルを用いることにした。

3.2 リアルタイム性

SPH は元々高速なシミュレーション手法であり、本研究により追加される計算量も少ないため、提案手法は十分なリアルタイム性を提供できる。

多くの場合、リアルタイム性が向上することにより、フィードバックが高速になるため、対話性も向上するが、格子法では空間制約の問題から対話性に制限が加えられている。粒子法によりシミュレーションを行うことができれば、ユーザ介入の自由度が増加する。特に気体では、液体と比べて、特定の空間内にとどまることが少ないため、この点は重要である。

4. 計算モデル

前章の要請を満たすため、SPH をベースとして用いた、シミュレーションの概要を図 2 に示す。化学反応による熱の発生を計算し、そこから発生した熱を SPH 相互作用により生じた力に、浮力項として加えることとした。すなわち、この浮力項をゼロとすれば（表面張力を除いた）流体シミュレーションと同一になる。

4.1 仮定

本研究では従来の SPH¹⁴⁾を拡張した形で扱うが、燃焼は複雑な物理・化学現象であり、これを厳密にシミュレーションする場合は、調整パラメータが増大してしまう。そこで、本

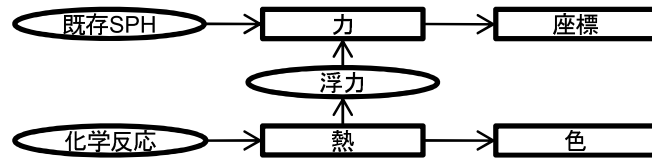


図2 シミュレーション概要．従来の粒子法にシンプルな化学反応モデルを追加し，浮力と熱を生成する
Fig. 2 Simulation overview. A simple chemical reaction model is incorporated into an existing particle method, so as to generate buoyancy and heat.

研究では次の仮定をおいた．

- 生成された熱量は直接温度に反映されるものとする．
- 酸素はつねに供給されるものとする．
- 燃料は気体燃料とする．

本章では，上記の仮定を離散化し，実現するモデルについて述べる． k_* と表記しているパラメータは，実際には1度定めてしまえば変更の必要がほとんどないものである．

4.2 基本モデル

SPH における離散化の基本式を解説する．任意の座標 \mathbf{r} における物理量 A は，

$$A(\mathbf{r}) = \sum_j A_j \frac{m_j}{\rho_j} W(\mathbf{r} - \mathbf{r}_j, h)$$

と計算される．ここで， A_j は粒子物理量， m_j は粒子質量， ρ_j は粒子密度， \mathbf{r}_j は粒子座標， h は粒子間影響距離， $W(\mathbf{r}, h)$ はカーネルである．

ナビエ・ストークス方程式の離散化には，Müller ら¹⁴⁾ と同様の計算モデルを用いる．以下ではこれを前提として説明する．

4.3 火炎粒子

本研究では扱う対象が火炎を含む気体であるため，各粒子に温度 T ，燃料 F を追加する．以下の条件を満たした粒子を火炎粒子であるとする．

- 温度が一定値より高い．
- 燃料を一定以上保持している．

4.4 浮力項

原理的には，温度に比例して周囲より密度が下がり，浮力が発生するのであるが，極端な密度差に対応することは粒子法，格子法ともに難しい．そのため，本研究では外力として温度に応じた浮力項をナビエ・ストークス方程式に加えることとした．すなわち，

$$\mathbf{f}_{buo} = k_{buo} (T - T_{amb}) \mathbf{z}$$

とする．ここで， k_{buo} は浮力係数であり， \mathbf{z} は上方向単位ベクトル， \mathbf{f}_{buo} は浮力項ベクトルである．また， T_{amb} はシミュレーション環境の温度であり，通常は 300 K 程度の定数となる．

4.5 熱拡散

熱拡散の取扱いには Müller ら¹⁵⁾ のモデルと同じものを用いる．すなわち， k_h を熱伝導係数として，

$$\frac{\partial T}{\partial t} = k_h \nabla^2 T$$

とモデル化し，これを離散化して，

$$\frac{\partial T_i}{\partial t} = k_h \sum_j \frac{m_j}{\rho_j} (T_j - T_i) \nabla^2 W(\mathbf{r}_i - \mathbf{r}_j, h)$$

とする．

4.6 燃料拡散

温度が高い状態ということは，燃料を近傍に放出しやすい状態と考えることができるため，燃料拡散係数を k_f として，

$$\frac{\partial F}{\partial t} = k_f T \nabla^2 F$$

とモデル化し，これを離散化して，

$$\frac{\partial F_i}{\partial t} = k_f \sum_j \frac{(T_j + T_i)}{2} (F_j - F_i) \frac{m_j}{\rho_j} \nabla^2 W(\mathbf{r}_i - \mathbf{r}_j, h)$$

とする．

4.7 化学反応

本研究では，火炎粒子が行う化学反応を以下のようにモデル化する．化学反応速度 S を

$$S = k_{rs} TF$$

で定義する．ここで k_{rs} は化学反応速度係数を表す．温度に応じた燃料を消費するため，

$$\frac{\partial F}{\partial t} = -k_{cf} S$$

とする．ここで， k_{cf} は燃料消費係数を表す．また，化学反応に応じて熱を発生するため，

$$\frac{\partial T}{\partial t} = k_{pt} S$$

とする．ここで， k_{pt} は温度生成係数を表す．

5. 実装

本研究では計算コストにも注意を払うため，効率的かつ効果的な実装を提示する．

5.1 近傍探索

近傍粒子探索においては，空間を均一セルに分割する手法¹⁸⁾ か，k-d 木を用いた手法¹⁾ が主に用いられる．

本研究の性質上，リアルタイム性を保持するために，比較的少ない粒子で，火炎の性質上，広大な空間を扱うことが必要となる．このため，空間を仮想化したソートによる近傍探索手法⁷⁾ を実装した．

5.2 リサンプリング

液体と違い，火炎のシミュレーションでは火炎の付近に粒子が配置されていなければ相互作用計算ができない．空間に粒子が充填されていれば高い精度での計算が可能となるが，大量の粒子はメモリ消費が激しく，また相互作用計算が増加するため，計算効率が低下してしまう．本研究では粒子の追加と削除によるリサンプリングを実装した．なお，同一ステップ内で粒子が増減すると条件分岐が増えてしまうため，追加，削除される粒子へのポインタを保持しておき，次のシミュレーションステップの先頭部で追加，削除する．リサンプリングは毎ステップ行うより，一定ステップおきにまとめて行う方が効率的である．本論文では5ステップおきに行った．リサンプリングの仕組みを図3に示す．

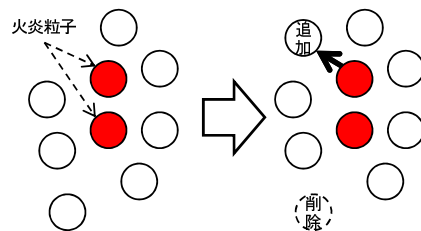


図3 リサンプリングの仕組み．各火炎粒子の法線方向に近傍に新しい粒子を生成する．各粒子の近傍に火炎粒子が存在していなければ，その粒子を削除する

Fig. 3 Re-sampling mechanism. New particles are generated around each flame particle using its normal vector. If there are no flame particles around a particle, it is deleted.

5.2.1 粒子の追加

火炎粒子の近傍に粒子を配置する必要があるが，単に周囲に粒子を追加したのではすでに存在している粒子と重複が発生してしまう．火炎粒子の近傍に，重複が発生しないように，粒子を配置するために，本研究では法線を用いた．法線は既存 SPH¹⁴⁾ の，表面張力を求める際に用いられる法線をそのまま用いることができる．火炎粒子の法線方向に粒子直径分移動させた座標に新しい粒子を配置する．この新しい粒子は燃料をゼロ，温度を環境温度として，300 K としておく．内部の火炎粒子には粒子の追加は不要であるため，近傍粒子数が一定未満の粒子の周囲にのみ追加を行う．これにより，衝突判定を行うことなく，高速に火炎近傍に粒子を充填することができる．

5.2.2 粒子の削除

近傍に火炎粒子が存在しない粒子を削除対象とする．具体的には近傍粒子ペア作成時に近傍火炎粒子をカウントしておき，ゼロであればその粒子を削除することとする．

5.3 固体粒子

シミュレーションの開始には火炎の燃焼源となる固体粒子が必要である．固体粒子はほかの粒子と同様に計算されるが，時間積分を行わないことによって自動的に速度，座標，燃料，温度が一定に保たれる．固体粒子の初期燃料と初期温度を燃焼可能値以上にしておけば，自動的に周囲にリサンプリングが行われ，火炎の生成源として扱うことが可能となる．この固体粒子から周囲の粒子に燃料と温度が拡散されるため，連鎖的に，かつ継続的に化学反応が行われることとなる．また，燃焼可能値未満の固体粒子を用意しておき，その固体粒子の速度をユーザにより指定させれば，火炎に固体からの1方向の作用を加えることが可能となる．なお，火炎粒子と固体粒子は，属性であり，粒子データ構造そのものは同一である．

5.4 擬似コード

シミュレーションプログラムを擬似コード1に示す．最初に従来の SPH と同様の計算を行ったのち，熱拡散，燃料拡散計算を行う．データ構造には，従来の SPH 粒子データ構造に対して，温度と燃料の追加が必要となる．本手法は，従来の SPH のプログラムに追加するだけで済む．

5.5 レンダリング

各粒子のレンダリングには，中心部の不透明度を高くしたテクスチャを貼り付け，ポイントスプライトを用いた．色は黒体放射による Nguyen らの手法¹⁶⁾ と同様の計算手法で定めた．この手法を用いて，3,000 K を最大温度とした色伝達関数を図4に示す．ただし実際の火炎においては温度にのみ依存する黒体放射だけでなく，炎色反応による色も大きな影響を

擬似コード 1 シミュレーションプログラム

```

while underSimulation do
  if isResamplingTimeStep then
    doResampling
  end if
  for all particles do
    calculateHeatDiffusion
    calculateFuelDiffusion
  end for
  for all particles do
    addBouyancyForce
    updateTemperature
    updateFuel
  end for
end while

```

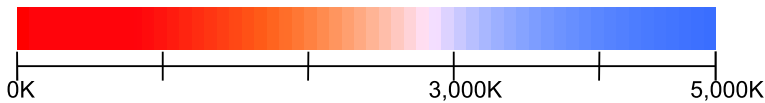


図 4 粒子の温度に対する色を決定する伝達関数．3,000 K を白色と仮定しており，低温になるにしたがって赤成分が多くなる

Fig.4 Color transfer function used for associating a color with the temperature of each particle. Color white is associated with the temperature of 3,000 K. Red component increases as the temperature decreases.

及ぼすが，燃料となっている物質まで考慮することはユーザの制御性を損ねてしまうため，本研究では取扱いを避けた．なお，火炎でない粒子はレンダリングしない．また，火炎の透明度に関する研究は見当たらなかったため，本研究では単純に不透明度を温度比例とした．すべての粒子を表示した場合，火炎粒子のみを表示した場合，ポイントスプライトを用いて炎をレンダリングした場合の比較を図 5 に示す．さらに，少数の粒子から，動的かつ滑らかな印象を生成するため，簡易的なモーションブラを利用する．具体的には，直前のレンダリング結果をバッファから読み出し，現フレームのレンダリング結果と一定比率で合成して

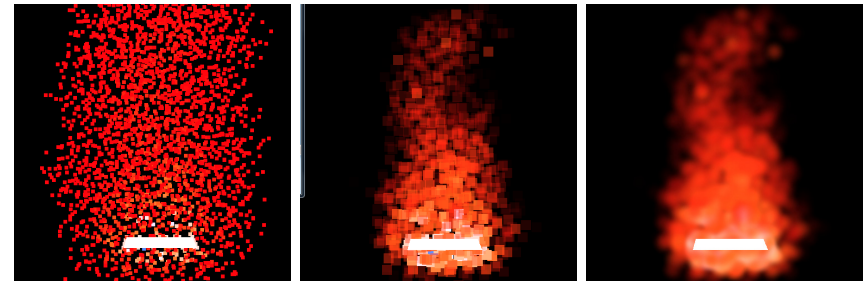


図 5 レンダリング比較．すべての粒子を表示した場合（左），火炎粒子のみを表示した場合（中央），火炎粒子にポイントスプライトを適用した場合（右）

Fig.5 Comparison of different rendering styles. All particles (left), flame particles (center), point sprites applied to flame particles (right).

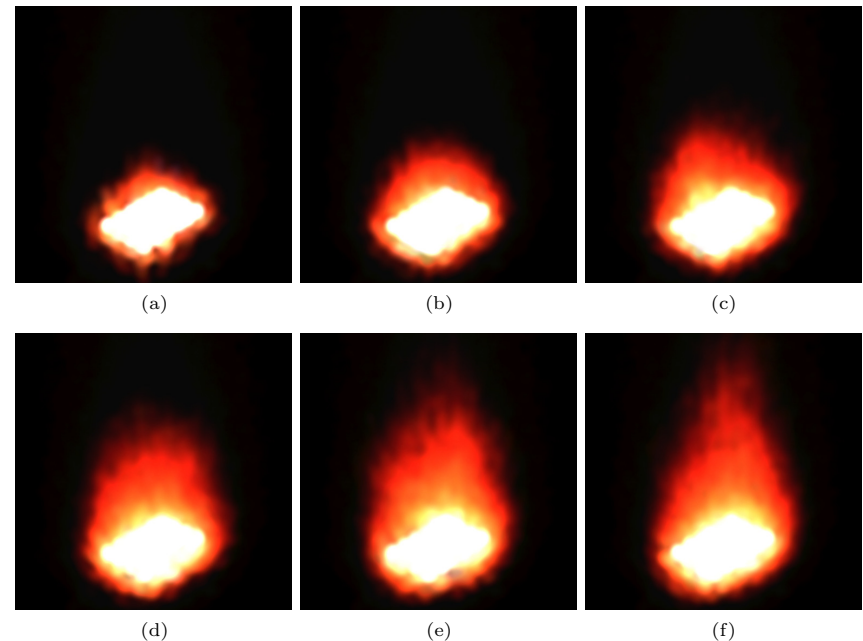


図 6 火炎の形成例．火炎の形成から安定状態への推移までが自動的に実現される

Fig.6 Flame formulation. Transition from initial to stable states is automatically generated.

表示すると同時に、再びバッファに保存する。

6. 結 果

本論文の結果は CPU に Intel Core2 2.40 GHz, 実装に C++, グラフィクス API に OpenGL を用いて実装したものである。安定状態を図 1 に、火炎の形成から安定状態への推移までを図 6 に示す。これらは、中心部が白色に近く、外側ほど赤みを帯びる点、火炎が上へと向かおうとする様子が表現されている。固体からの作用を受けるシミュレーションを図 7 に示す。図中の白板が固体である。火炎の形状が固体からの片方向の作用を受けて、変化していく様子が確認される。

また、外力を加えた例を図 8 に示す。強い外力を受けている際には火炎が小さくなっていく

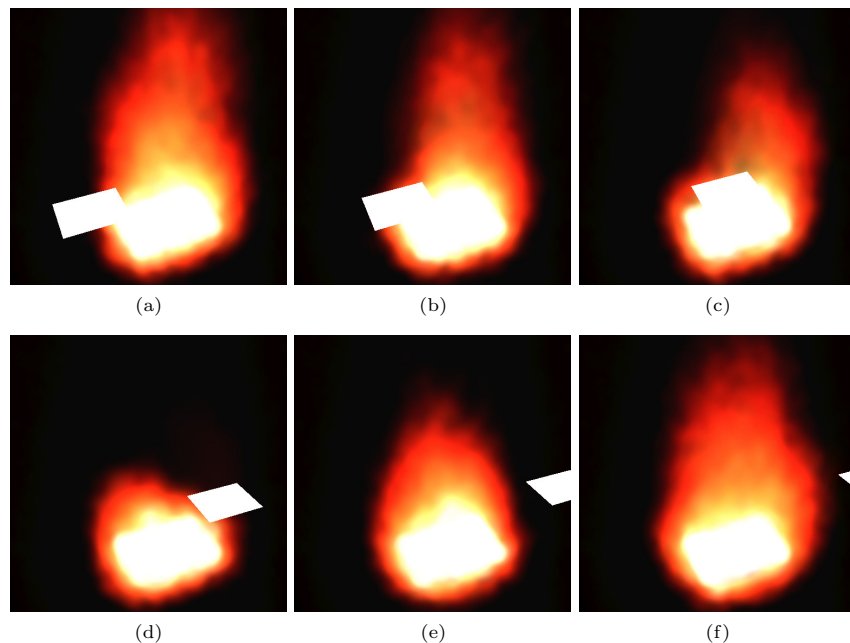


図 7 固体からの作用例。固体が火炎を通過している間、火炎は小さくなり、通過後に再び火炎が大きくなる
Fig. 7 One-way coupling with solid. The flame gets smaller against the passing solid, but it grows afterwards.

る点特徴的である。これは外力により熱が逃げているためと考えられる。これらは乱数とライフタイム（粒子が存在する期間）を用いる従来の粒子法¹⁷⁾では困難であり、物理ベースならではのシミュレーションと考えられる。主には固体粒子の座標と、その温度、燃料、また外力を調整するだけでこれらの多様なシミュレーションを生成することができ、ユーザの制御性も高いものと考えられる。これらのシミュレーションは粒子数 2,500 前後で安定し、レンダリングを含めて約 40 fps で実行された。

さらに、火炎のマージ例を図 9 に示す。2 つの火炎がマージされ、1 つの大きな火炎となっていることが確認できる。格子法と違い、事前にシミュレーション空間を定義する必要がないため、動的な変化への対応が容易である。また、従来の粒子法と違い、ランダムに動くのではなく、自然と一体化されていく様子が確認できる。このシミュレーションは粒子数 5,000 前後で安定し、レンダリングを含めて約 20 fps で実行された。

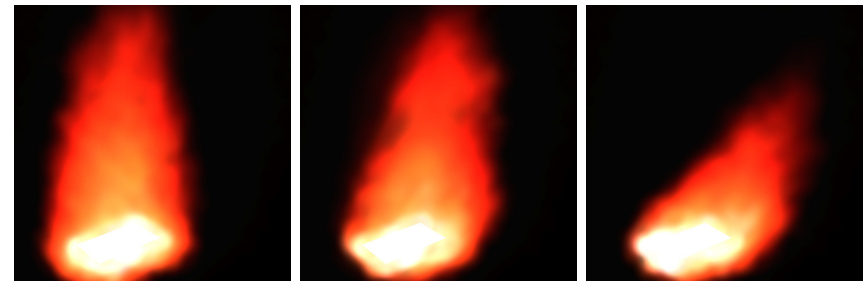


図 8 外力による作用例。強い外力を受けている間、火炎はなびき、小さくなる
Fig. 8 Effect of external force. The flame gets smaller by a strong force.

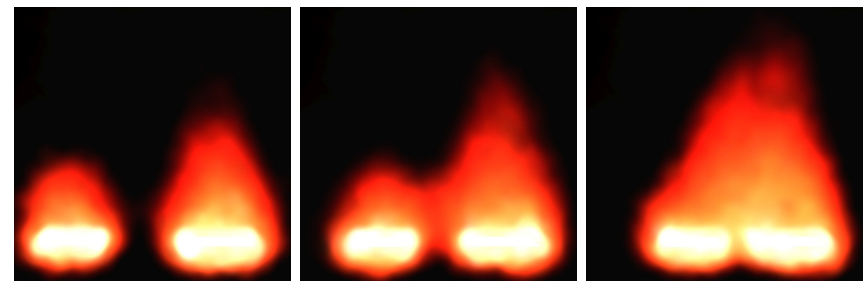


図 9 火炎のマージ例。2 つの火炎が滑らかに一体化し、1 つの大きな火炎となっていく
Fig. 9 Flame merging. Two flames merge smoothly into a big one.

7. 結論と今後の課題

本論文では既存 SPH の拡張によるリアルタイムな火炎シミュレーション技法を提案した。結果として、従来の粒子法では難しい火炎の形成から安定状態の形成、固体からの作用による影響までをリアルタイムにシミュレーションすることができた。

今後のシミュレーションの課題としては、火炎そのものの表現としては、GPU を用いて、リアルタイムに使用可能な粒子数を増やし、乱流的要素を表現することや、火炎以外の表現として、すすや煙の付加などがあげられる。

また、本論文では、レンダリングよりシミュレーションに重点を置いたが、少ない粒子によるディテールの表現、揺らぎの表現など、レンダリングの改良も重要である。

参 考 文 献

- 1) Adams, B., Pauly, M., Keiser, R. and Guibas, L.J.: Adaptively Sampled Particle Fluids, *ACM Trans. Graphics*, Vol.26, No.3, Article 48 (2007).
- 2) Desbrun, M. and Gascuel, M.: Smoothed Particles: A New Paradigm for Animating Highly Deformable Bodies, *Proc. Eurographics Workshop on Computer Animation and Simulation*, pp.61–76 (1995).
- 3) Enright, D., Marshner, S. and Fedkiw, R.: Animation and Rendering of Complex Water Surfaces, *ACM Trans. Graphics*, Vol.21, No.3, pp.736–744 (2002).
- 4) Fedkiw, R., Stam, J. and Jensen, H.: Visual Simulation of Smoke, *Proc. ACM SIGGRAPH 2001*, pp.15–22 (2001).
- 5) Foster, N. and Fedkiw, R.: Practical Animation of Liquids, *Proc. ACM SIGGRAPH 2001*, pp.15–22 (2001).
- 6) Foster, N. and Metaxas, D.: Realistic Animation of Liquids, *Graphical Models and Image Processing*, Vol.58, No.5, pp.471–483 (1996).
- 7) Green, S.: Particle-Based Fluid Simulation, *Game Developers Conference (2008)*, available from (http://developer.download.nvidia.com/presentations/2008/GDC/GDC08_ParticleFluids.pdf).
- 8) Harada, T., Koshizuka, S. and Kawaguchi, Y.: Smoothed Particle Hydrodynamics on GPUs, *Proc. Computer Graphics International*, pp.63–70 (2007).
- 9) Hong, J.M., Shinar, T. and Fedkiw, R.: Wrinkled Flames and Cellular Patterns, *ACM Trans. Graphics*, Vol.26, No.3, Article 47 (2007).
- 10) Horvath, C. and Geiger, W.: Directable, High-Resolution Simulation of Fire on the GPU, *ACM Trans. Graphics*, Vol.28, No.3, Article 41 (2009).
- 11) Ihm, I., Kang, B. and Cha, D.: Animation of Reactive Gaseous Fluids Through Chemical Kinetics, *Proc. 2004 Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, pp.203–212 (2004).
- 12) Inakage, M.: A Simple Model of Flames, *Proc. Computer Graphics International*, pp.71–81 (1990).
- 13) Melek, Z.: Interactive Simulation of Fire, Burn and Decomposition, Ph.D. Dissertation, Texas A&M University (2007).
- 14) Müller, M., Charypar, M. and Gross, M.: Particle-Based Fluid Simulation for Interactive Applications, *Proc. Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, pp.154–159 (2003).
- 15) Müller, M., Solenthaler, B., Keiser, R. and Gross, M.: Particle-Based Fluid-Fluid Interaction, *Proc. Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, pp.237–244 (2005).
- 16) Nguyen, D., Gibou, F. and Fedkiw, R.: Physically Based Modeling and Animation of Fire, *ACM Trans. Graphics*, Vol.21, No.3, pp.721–728 (2002).
- 17) Reeves, W.T.: Particle Systems – A Technique for Modeling a Class of Fuzzy Objects, *ACM Trans. Graphics*, Vol.2, No.2, pp.91–108 (1982).
- 18) Teschner, M., Heidelberger, B., Müller, M., Pomeranets, D. and Gross, M.: Optimized Spatial Hashing for Collision Detection of Deformable Objects, *Proc. Vision, Modeling, and Visualization*, pp.47–54 (2003).
- 19) Zhu, J., Liu, Y., Bao, K., Chang, Y. and Wu, E.: Realtime Simulation of Burning Solids on GPU with CUDA, *Proc. 2010 10th IEEE International Conference on Computer and Information Technology*, pp.1219–1224 (2010).
- 20) 井村誠孝, 稲垣 智, 池田 聖, 眞鍋佳嗣, 大城 理, 千原國宏: 粒子ベース流体シミュレーションに基づく炎のリアルタイムレンダリング, 第 15 回日本バーチャルリアリティ学会大会論文集, pp.560–563 (2010).
- 21) 間淵 聡, 藤代一成, 大野義夫: SPH ベース火炎シミュレーション, 情報処理学会研究報告, Vol.2010-CG-140, No.9 (2010).

(平成 23 年 2 月 18 日受付)

(平成 23 年 7 月 8 日採録)

推薦文

本論文は、粒子ベースの SPH 法を利用して火炎をシミュレートする手法を提案しています。この手法は化学反応によって発生する熱を浮力項として加えるという、従来の SPH 法に簡便な拡張を施すことで、インタラクティブに 3 次元シミュレーションを実現することを可能にしています。これは情報処理学会会員にとっても興味深くかつ有益な研究であり、論文誌での掲載価値を十分に有していると判断し、本研究会は本論文を推薦論文として推薦するものです。
(グラフィクスと CAD 研究会主査 山口 泰)



間淵 聡 (学生会員)

2005 年東京工科大学工学部卒業。2007 年慶應義塾大学大学院理工学研究科修士課程修了。CAD/CAM 会社勤務を経て、現在、慶應義塾大学大学院理工学研究科博士課程在籍中。粒子法によるシミュレーションとレンダリングについて研究を行っている。



藤代 一成 (正会員)

1985 年筑波大学大学院博士課程工学研究科修士号取得退学後、東京大学、筑波大学、お茶の水女子大学、東北大学を経て、2009 年より慶應義塾大学理工学部情報工学科教授。理学博士 (東京大学)。ポリウムグラフィックス、可視化設計支援、多感覚情報呈示に関する研究に従事。IEEE Computer Society, ACM, Eurographics ほか会員。



大野 義夫 (正会員)

1973 年慶應義塾大学大学院工学研究科博士課程単位取得退学。1970 年同大学情報科学研究所助手。その後専任講師、助教授を経て 2011 年まで同大学理工学部教授。現在は東北公益文科大学特任教授。コンピュータグラフィックスおよび形状モデリングの研究に従事。芸術科学会ほか会員。