

# 論文内容の要旨

論文題目 **Controlled, Multiple-Scale Destruction Simulation with Fast Rendering Techniques**

(制御された多重スケール破壊シミュレーションとその高速レンダリング)

氏名 今給黎 隆

## 1. はじめに

物体の衝突に伴う破壊シミュレーションは、コンピュータグラフィックスの分野において興味深いテーマの1つになっている。破壊シミュレーションは、映画やゲームなどにおいてリアルなアニメーションを生成するためには欠かすことができなくなりつつある。しかしながら、破壊シミュレーションは、元の物体が複数に分割されるため、剛体シミュレーションと比較して処理すべき計算量が増え、現実に近いリアルな現象を思うように作るのは、未だに困難である。

従来の物理ベースのシミュレーションでは、破壊は物体を複数の破片に分解することで表現されてきた。しかし、現実の現象を観察すると、細かな破片や粉煙が発生しており、従来法ではこれらを高速に再現することができなかった。特に物体の衝突に伴う破壊における粉煙の生成は全く考慮されてこなかった。

別の視点に目を向けると、映画などでは、あらかじめ絵コンテによりシーンが設計されるため、通常では発生し難い演出をしたいことがしばしば発生する。しかし、通常の物理シミュレーションでは、運動は決定論的に決まり、望む結果が得られるように初期値やそれらの特性を指定することは難しい。

そこで、本研究では、細かな破壊の表現として様々な大きさの破片と粉煙を生成する統一的な破壊シミュレーションを提案する[1]。提案法では、粗いスケールの破砕、細かな破片と粉煙の3種類の異なる階層のシミュレーションを導入する。粗いスケールの破砕のシミュレーションとして、従来法として知られる、拡張個別要素法(Extended Distinct Element Method, EDEM)[3]を改良する。物体が破壊された際の各小片に関して、破壊の際のエネルギーから大きさの分布を計算する。最終的に、物体が衝突する際の破壊に関して、さまざまな大きさの破片を生成する物理シミュレーションを実現する(図1)。

また、物体の破壊後の形状を制御する手法を提案する[2]。EDEMでは、物体を要素(破片)の集合で表現し、要素間にばねを設置する。提案法では、このシミュレーションにおいて、ユーザーにより指定された物体の堅い部分と脆い部分の情報を元に要素間のばねの切れ難さを個別に

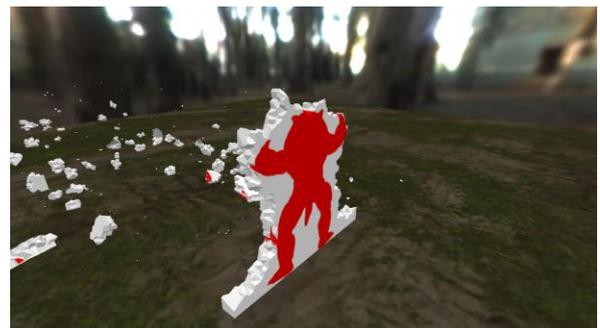
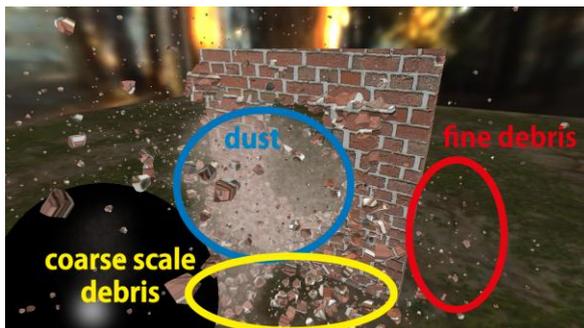


図1: 提案法による3種類の物理シミュレーションの結果。

図2: 破壊時の亀裂の形状の制御の結果。

黄, 赤, 青の円の中に各スケールで特徴的な結果が得られる。赤い部分が壊れ難い。



図 3: スライス法によるボリュームレンダリング(左)と、その拡大図(右)。エイリアシングが発生する。 図 4: 提案法によるボリュームレンダリング(左)と、その拡大図(右)。

構築することで、破壊後の形状を制御する。ユーザーは物理的な運動をしつつも、意図した形状の破砕物を生じるアニメーションの生成が可能となる(図 2)。

実際のアプリケーションでは、実行速度も重要な要素の 1 つである。ゲームでは、リアルタイムな応答性が要求される。また、映像制作においても、高速に画像を作成することで作業の修正および確認の回数を増やすことが可能となり、完成する映像の品質が向上する。粉煙のレンダリングは、実行速度のボトルネックになりやすい処理の 1 つである。スライス法は、粉煙の濃度分布を半透明な媒質と捕らえてレンダリングする技法であり、幅広く用いられている。しかしながら、処理を高速化するには、スライス面の数を少なくせざるを得ない。その結果、レンダリング画像には縞模様のエイリアシング(縞模様)が現れる(図 3)。本研究では、半透明物体を含むシーンに関して、レンダリング結果にエイリアシングが生じない高速な手法を提案する[4]。提案法では、半透明物体とポリゴンによる物体が接した際の境界にエイリアシングが発生しない(図 4)。本手法は、GPU を用いて効率的に実装できる。

## 2. 多重スケールの破壊シミュレーション

本研究では、物体に力学的な衝撃を加えて破壊するシーンを取り扱う。まず、粉煙を含む細かな破片を伴う破壊現象を高速にシミュレーションする手法を提案する。破壊現象の幅広い表現を行うために異なるスケールを持つ 3 種類の物理シミュレーションを使用する。

1 つ目は、EDEM であり、粉砕に要するエネルギーの計算や粗いスケールの破壊の表現を扱う。EDEM では、衝突する物体を小さな要素の集合で構築し、要素の運動

$$M \frac{dv_I}{dt} = \sum_{J \in I \text{ と 接触}} F_{JI}^c + \sum_{K \in \text{間隙ばねで接続}} F_{KI}^p + Mg, \quad \frac{dx_I}{dt} = v_I,$$

$$F_{JI}^c = -k^{cn} \Delta x_{JI}^{cn} - \eta^{cn} v_{JI}^{cn} - \eta^{ct} v_{JI}^{ct}, \quad F_{KI}^p = -k^p \Delta x_{KI},$$

の集合として系の運動を計算する。ここで、 $\Delta x_{JI}^{cn}$  は要素  $\{I, J\}$  間におけるめり込み量に関して要素の中心の差分ベクトルの方向(法線方向)の成分、 $v_{JI}^{cn}, v_{JI}^{ct}$  は要素間の相対速度の法線方向および法線方向に垂直な成分、 $\Delta x_{KI}$  は要素  $\{I, K\}$  間に設けられたモルタル等の間隙材をモデル化する間隙ばねの初期状態からの変位ベクトルである。破砕現象は、間隙ばねによる要素間の結合として計算する(図 5)。

2 つ目は、粒子シミュレーションであり、細かな粒度の破砕物の表現を行う。間隙ばねのばねエネルギーを計算し、蓄えられたエネルギーで物体を破砕した際の破砕物の最大の大きさを計算する。この大きさが EDEM の各要素の大きさよりも小さい場合に、要素は砕かれたと捕らえる。砕かれた要素はその位置や速度を基に小片を追加する。その際に、現実の破壊において経験的に広く知られる破砕物の大きさの分布式である Gaudin - Schuhmann 分布を仮定して、各小片の大きさを決定する。粒子シミュレーションでは、小片ごとの衝突判定を省略し、他の物体との衝突処

理について大きさを持たない点として処理することで高速化を図る。

3つ目は、流体シミュレーションである。EDEM で要素に加えられたひずみエネルギーの大きさから、細かな破砕物が生成される量を推定し、流体シミュレーションへ粉煙の濃度を与える。

レンダリングに関しても3種類の異なるスケールを扱う。1つ目は、3次元のボロノイ領域を基本単位とするボロノイ図を利用する。EDEM の各要素の中心を母点とするボロノイ図を計算する。ボロノイ図は前計算で作成し、シミュレーション中の細分割は行わない。メッシュの描画では、各頂点に関して、それぞれのボロノイ領域における相対座標が必要となる。ボロノイ領域の位置や向きは、複数の頂点から参照されるため、ボロノイ領域の位置や向きをテクスチャとしてビデオメモリに転送する。また、メッシュの頂点を含むボロノイ領域に対応する要素同士を繋ぐ間隙ばねが消失しているのか、各時刻で判定する必要がある。メッシュの頂点数と同じテクセル数のテクスチャを用意し、各テクセルを頂点に対応させ、テクスチャのRGBAの4成分に対して、頂点に対応する要素を繋ぐ間隙ばねが消失したか記録することで GPU による効率的な実装を行う。2つ目のレンダリングは、小片のレンダリングである。EDEM の要素のメッシュを縮小して表示し、効率的にレンダリングする。3つ目は、粉煙の表現に煙のレンダリング手法を用いる。

### 3. 破壊シミュレーションにおける亀裂の形状の制御

ユーザーによる破壊された物体の形状の制御について取り扱う。本研究では、間隙ばねが切れる長さや曲げおよびひねりの角度の閾値を個別に設定することで、亀裂の制御を行う。破壊後の物体の形状の指定に「脆さマップ」と名付ける画像を用いる。脆さマップで指定された物体の脆さ情報から、間隙ばねを階層的にグループ化する。グループ化には、破碎のために加えられた力が大きな際には小さな破片が生じ、破碎のための力が小さな際には大きな破片のみが発生するように、間隙ばねの要素の空間的距離に応じてグループ分けを行う。破壊物の形状は、階層ごとに間隙ばねの切れ難さの閾値に係数  $p_i$

$$p_i = p_0 + (p_H - p_0)(h_i / H)^n,$$

を乗算することで、切れ難さを制御する。ここで、 $h_i$  は要素  $i$  の階層の値、 $H$  は階層の最大値、 $p_0$ 、 $p_H$  及び  $n$  は、調整のためのパラメータである。ばねのグループ分けの際に、物体の破壊したい形状を考慮することで、ユーザーが指定した形状に近い形に物体が壊れる制御を実現する。

なお、階層化を用いない場合には、提案法による結果(図2)と異なり、破壊される形状の制御は十分には行えず、壊れて欲しくない部分も破壊されやすくなる(図6)。

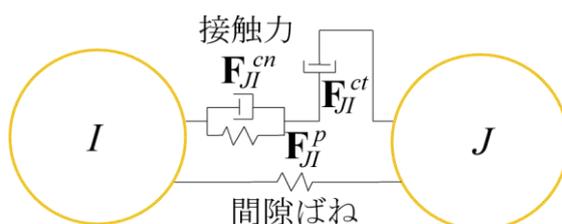


図5: EDEM 要素間に働く力。



図6: 階層化を用いず、脆さマップの色に応じてばねの切れ難さを設定した例。図2と比較すると、本来崩れて欲しくない腕の領域も切り取られている。

#### 4. ボリュームレンダリングの高速化

半透明な媒体の効果をリアルタイムに描画する手法として、スライス法が知られている[5]。スライス法は、濃度分布を持つボリュームデータの断面であるスライス面を複数用意し、奥から並べて描画することでボリュームを可視化する手法である。スライス法で高速にレンダリングするためには、スライス面の枚数を少なくする必要がある。しかし、スライス面の枚数が少ない場合には、ボリュームの本来の形状との差が大きくなり縞模様のエイリアシングが発生する。

提案法では、スライス面の代わりに凸包のプリミティブ(図7)を描画し、その内部の複数のボリュームデータをサンプリング間隔を調整しながら読み込む。その際に、不透明物体の深度を参照することで、不透明物体とのエイリアシングも取り除く。提案法は、他の物体がボリュームに対して落とす影に関して、複数の値を補間することでエイリアシングのない画像を生成することもできる(図8)。

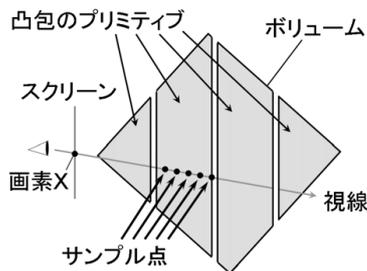


図7: 凸包のプリミティブ。



図8: 提案法のスポットライトの光跡への適応例。

#### 5. まとめ

幅広い破壊シミュレーションの手法として、3種類のシミュレーションを使う手法を提案した。提案法では、粗いスケールのシミュレーションに関して、ポロノイ図から作成されたメッシュをGPUで高速にレンダリングすることが可能となった。また、破壊時に生じる亀裂の形状を指定する手法を提案した。破片を繋ぐばねに関して、空間的な距離による階層化を用いて切れ難さを設定することで、ユーザーが指定した形状に近い形に物体が壊れる制御を実現する。さらに、粉煙をレンダリングするボリュームレンダリングに関して、凸包のオブジェクトを導入し、その内部の領域を視線方向に複数回サンプリングすることで、高速にレンダリングする手法を提案した。これらの方法により、インタラクティブなアプリケーションに関して、高速でユーザーが制御可能な破壊シミュレーションが可能となった。

#### 参考文献

- [1] 今給黎 隆, ヘンリー ジョハン, 西田 友是, 「破壊後の形状の制御を伴う破壊シミュレーション」, 画像電子学会, 画像電子学会 VC 特集号, 2009.
- [2] Takashi Imagire, Henry Johan, Tomoyuki Nishita, "A Fast Method for Simulating Destruction and the Generated Dust and Debris," *The Visual Computer*, 25, 5-7, pp.719-727, (2009).
- [4] Meguro K., Hakuno M., "Fracture analyses of concrete structures by the modified distinct element Mmethod," *Structural Eng./Earthquake Eng.*, JSCE 6, 2, pp.283-294, 1989.
- [3] Takashi Imagire, Henry Johan, Naoki Tamura, Tomoyuki Nishita, "Anti-Aliased and Real-Time Rendering of Scenes with Light Scattering Effects," *The Visual Computer*, 23, 9-11, pp.935-944, (2007).
- [5] T. J. Cullip, U. Neumann, "Accelerating volume reconstruction with 3D texture hardware", Tech. Rep. TR93-027, University of North Carolina, Chapel Hill N.C..