

砂塵の対話的ビジュアルシミュレーション

中島 聡 大野 義夫 藤代 一成
慶應義塾大学理工学部情報工学科

1 背景と目的

近年、映画やゲームで、自然現象のシミュレーションをコンピュータグラフィックスで描いたシーンをよく見かける。その対象の一つに、砂の動きが挙げられる。しかし、砂と水とのインタラクションや、砂と固体とのインタラクションを扱った研究は報告されているが、砂と風とのインタラクションを扱ったものは見当たらない。

そこで本研究では砂塵を対象として、ビジュアルシミュレーションを実現するアルゴリズムを提案する。ここでは、風を発生させて砂塵を表現し、これが地形により受ける影響を考慮しながら、ユーザが自由に砂塵を移動させることができることを目的としている。実世界では、砂は物理的に様々な力を受けて複雑な動きをするが、本研究では物理的に厳密であるよりも、見た目にシミュレーションできていることや、砂塵の移動のインタラクティブ性に重点をおいている。

2 砂塵の表現

本節では、砂塵を表現するために用いたアルゴリズムについて説明する。

2.1 風の発生

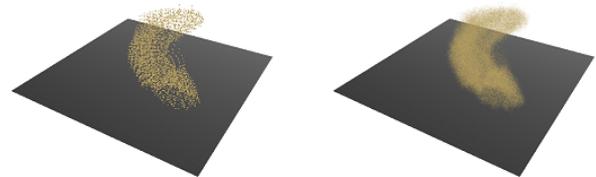
まずある大きさのベクトル場空間を作り出す。このベクトル場は格子点ごとにベクトルを発生させている。粒子の現在の位置から一番近い格子点のベクトルに沿って粒子を動かすことで、擬似的に風を表現した(図1(a))。今回はこのベクトル場空間の生成に、Crawfisのトルネード生成プログラムを利用している。このプログラムでは、トルネードの渦中心線を求め、そこからの距離などにより、各格子点のベクトルを求めている。

2.2 風のランダム性

ただベクトル場を発生させるだけでは、定常的な砂塵になってしまう。しかし実際の砂塵は一定ではなく、ランダムに形を変えて動いている。そこで現実の動きに近づけるために、砂塵の渦中心線自体を回転させた。各タイムステップに回転させると粒子が渦の中央に集まってしまうので、一定時間ごとに上から見て時計回りに回転させるようにし、この回転角をある範囲でランダムにすることで、実際の砂塵に近い表現を実現した(図2)。

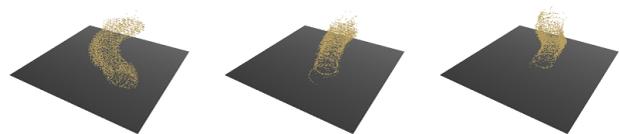
2.3 レンダリング

砂塵の煙に近い表現を実現するために、粒子ボリュームレンダリング(Particle-based Volume Rendering: PBVR) [1] を利用した。まず実際の粒子の周りに、擬似粒子をランダムに発生させる。それをZバッファ法を用いて投影し、その画像をアキュムレーションバッファに保存する。これを何回か繰り返し、保存した画像の輝度を足し合わせていく。最後に、足し合わせた画像の輝度を繰り返し回数で割る。これにより、粒子数の少ない部分は薄くなり、煙のような表現を可能にした(図1(b))。PBVRを行う場合は、実際の粒子は描いても薄くならないので、描かない。



(a) PBVR を行わない場合 (b) PBVR を行った場合

図1: 粒子ボリュームレンダリング(PBVR)を行った場合と行わない場合のレンダリングの比較。PBVRはランダム粒子を2個、繰り返し回数を16回とし、描いた粒子数は(a)が約2,500個、(b)が約5,000個である。



(a) 回転経過 1 (b) 回転経過 2 (c) 回転経過 3

図2: 回転角にランダム性を加えた結果

3 地形

本節では地形の生成手法と、地形の影響で風がどのように変化するかを考える。

3.1 地形の生成

地形は、中点変位法を用いてハイトフィールドを作成し、それをポリゴン表現することで、生成した。図3(a)のようなテクスチャを貼り付け、頂点法線ベクトルを用いて陰影付けを行うことで、砂漠を表現した。



図 3: 生成した地形

3.2 地形の影響

次に地形の影響について考えた。風は地面に向かって吹いたとき、坂に沿うように変化すると考えた。そこで、風速度ベクトルが地面方向に向かっているとき、なおかつ次のタイムステップでの粒子の位置が地面よりも下になるときに、風速度ベクトルを坂に沿うように修正させる処理を行う。地形の表面ベクトルと法線ベクトルの内積は0である。これを利用して以下の式(1)を導く。

$$\mathbf{r} = \mathbf{v} - \frac{\mathbf{n} \cdot \mathbf{v}}{|\mathbf{n}|^2} \mathbf{n} \quad (1)$$

ここで、 \mathbf{r} は修正後の速度ベクトル、 \mathbf{v} は速度ベクトル、 \mathbf{n} は地面の法線ベクトルである。このようにして、地面に平行なベクトルに修正することで、砂塵が坂に沿うようにした。今回は、粒子が地面から離れていく場合には地面の影響は無視している。

4 砂塵の移動

移動を可能にするために、地形を表現するワールド座標系とは別に、砂塵を発生しているベクトル場空間を考えた。この空間内の粒子の位置を求め、そこにマウスでドラッグした分の距離を足して空間ごと動かすことでワールド座標系での相対位置を計算し、そこに粒子を移動させた。

ユーザが砂塵のシミュレーションを観察しやすいように、左画面と右画面の二つのウィンドウを用いた。左画面が真上から見たもので、右画面が斜めから見たものになっている。砂塵を移動させるには左のウィンドウを用いて、右は視点を変えるためのものである。左のウィンドウでマウスをドラッグすることで、砂塵を移動させることができる。なお、移動させる空間がわかりやすいように、空間の端の点と中心の点を赤い球で描いている。

5 実験結果

開発言語に Visual C++ を用いて実装した。実行環境は Windows 7 (CPU: Intel(R) Xeron(R) 2.53GHz, メモリ: 12.0GB) である。ポリゴンサイズは 63×63 で水平方向の格子点間隔は 2, 元の粒子数は 3,500 ~ 5,000 個に対して描く粒子数は 7,000 ~ 10,000 個、アキュムレーションバッファに格納する画像の解像度は 640×480 で 1 タイムステップに 16 枚として、13 ~ 17 FPS(Frame per Seconds) であった。シミュレーション結果に PBVR を適用し、地形の影響も考慮して、砂塵を移動させている場合の画像を図 4 に示す。

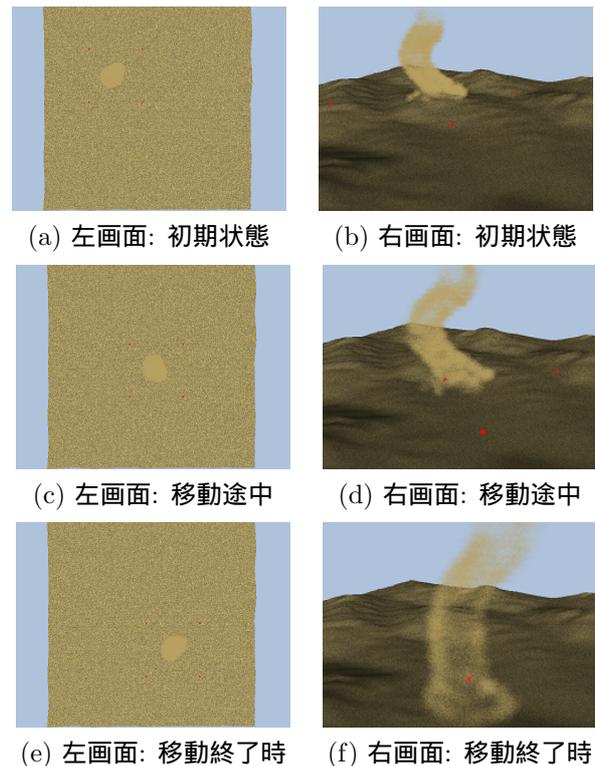


図 4: シミュレーション結果。ユーザが左画面の左上から右下に砂塵を移動させている。

6 まとめと今後の課題

本稿では、砂塵を表現し、これをユーザが自由に動かし、ビジュアルシミュレーションを行うためのアルゴリズムを提案した。

今後の課題としては、まだつむじ風でしかビジュアルシミュレーションを行えていないことが挙げられる。それ以外にも、ユーザが自由に風を発生させ、操ることができれば、より多様なビジュアルシミュレーションを行えると考えられる。レンダリングも、今回は PBVR を利用したが、近傍粒子から微小面を抽出し、その法線情報を用いることで、立体感をもった砂塵を表現することが可能になる。また、砂塵が通ったときに地面を削っていくような表現や、砂塵の通り道に障害物がある場合、さらには摩擦や空気抵抗のような物理的な厳密さなどを考慮すれば、さらにリアリティのあるビジュアルシミュレーションができると考えられる。

謝辞

風の生成コードの一部は、米国オハイオ州立大学の Roger Crawfis 准教授から提供されたものである。

参考文献

- [1] 小山田 耕二, 坂本 尚久: 粒子ポリウムレンダリング 理論とプログラミング, コロナ社, 2010 年 10 月。