

スケッチ入力を用いた氷・霜アニメーション生成のための 初期値最適化による成長速度制御

天野 幹子^{1,a)} 佐藤 周平^{2,b)} 伊藤 貴之^{1,c)}

概要: 本研究では、映像制作に欠かせないビジュアルエフェクツの表現の中でも氷や霜がオブジェクトに張り付いて広がる様子を表現するスケッチツールを提案する。平面のオブジェクト上においてユーザは広がり範囲と始点をスケッチにより入力し、その入力に従ってシミュレーションを行う。その際、従来のシミュレーション方法に乱数を導入することで、様々な広がり表現を可能にする。また、最急降下法によるシミュレーションの初期値の最適化を行うことで、ユーザの所望の速度での広がりを実現する。これにより、始点から指定した範囲内にユーザの所望の速度で氷・霜が樹枝状に広がるアニメーションが作成できる。

キーワード: 氷の形成、樹枝状成長、手続き型アニメーション、最急降下法

1. はじめに

3次元コンピュータグラフィックス（以下3DCGと称する）を用いた映像がメジャーになり、ビジュアルエフェクツ（以下VFXと称する）は映画やコンピュータゲームなどに広く適用されている。自然現象を実際のように再現するアニメーションによって、様々なシーンにおいてリアリティを高めることができる。近年ではVFXの作成をサポートするために、多くのソフトウェアやツールが開発されている。

氷と霜の結晶化は、VFXの中でも重要な自然現象の1つである。いくつかのシミュレーション方法 [1-3] が3DCGにおける氷や霜の形成手法として提案されている。これらの方法は氷や霜の形成のリアルなアニメーションを作成できる。しかしこれらの方法では、ユーザが所望の映像を作成可能とすることには焦点を当てておらず、それを実現するためにはシミュレーションのパラメータを試行錯誤的に調整する必要がある。これにはシミュレーションの実行とパラメータの調整作業を何度も繰り返す必要があり、多くの時間と労力を要する。この問題に対し我々は、ユーザが望む形状の氷の形成アニメーションを、スケッチ入力を用

いて直観的に作成可能とすることを目標に研究を行っている [4]。本稿では、氷や霜の成長のランダム性を表現できるように、シミュレーションの拡張を行う方法について報告する。ユーザはパラメータの調整を行うのみで、シミュレーションによって多様な広がりを作り出すことができ、ユーザの表現の幅が広がる。また、ユーザの所望の速度での氷・霜の広がりを実現するため、最急降下法によるシミュレーションの初期値最適化を行う。これによりユーザは細かいパラメータ調整の必要なく、目標のフレームでアニメーションが終了するような初期値でシミュレーションを行うことができる。本手法では、平面上の格子点を用いてシミュレーションを行うものとする。そして新たなパラメータの導入により霜のランダムな広がりを表現・制御し、かつその成長速度を初期値最適化により制御可能とした。

2. 関連研究

結晶の樹枝状成長: 氷や霜の結晶の視覚的な形状の特徴として樹枝状構造があげられる。氷や霜の広がりを表現するにあたってこの特徴は非常に重要であり、この構造を再現することによって本研究で表現するVFXが特徴付けられると考える。Kim [1] が考察している通り、氷の結晶の形は成長の際の大気の状態によっても異なり、結晶の幾何学構造は非常に複雑である。ほとんどの氷の結晶が樹枝構造と扇形板の連続で構成されていると考察でき、それらの連続によって多くの種類の結晶構造が生まれる。また、霜の幾何学的特徴として、境界付近に樹枝状の成長がみられ

¹ お茶の水女子大学
Ochanomizu University

² 富山大学
University of Toyama

a) g1520502@is.ocha.ac.jp

b) ssato@eng.u-toyama.ac.jp

c) itot@is.ocha.ac.jp

る。本報告で適用する従来手法 [2] は、六方向に伸びる氷の結晶成長を表現するものであり、それらが結合した霜のようなランダムな成長を表現するものではない。我々は乱数を導入することにより、ランダム性の表現を可能にする。

結晶成長表現のためのスケッチインタフェース：氷や霜の結晶を 3DCG で表現する研究は既にいくつか発表されている。その中でもスケッチ入力を用いるものとして Lipton らの手法 [5] があげられる。この手法ではアーティストが広がりや速度と形状を色ごとに分けてペイントする (図 1 参照)。その形状に沿って拡散する点群 (以下ポイントと称する) について、非構造化セルオートマトンのシミュレーションにより速度場を生成し、この速度場を用いてポイント動かした軌跡をエフェクトの線として用いている。

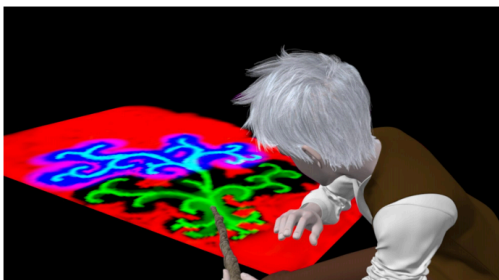


図 1 Lipton らの手法 [5] による氷・霜のデザイン。

3. 提案手法

本研究では、手続き的なシミュレーション方法を採用することにより、単純なスケッチ入力と直感的なパラメータの調整のみで氷・霜の広がりを表現する。まず、ユーザのスケッチ入力を基にシミュレーション領域を設定する。そして、Reiter の手法 [2] を用いてシミュレーションを実行する。本稿では Reiter の手法に乱数を導入することで、氷の広がりにより多様性を持たせられるよう拡張する。また、ユーザは任意の最終フレームを指定し、そのフレームに氷がシミュレーション領域の端に到達するよう最急降下法によりパラメータを最適化することで、所望の速さでの広がりを実現する。現在は、3D のオブジェクト上にユーザがペイントしたものを展開したと仮定し、二次元の任意の領域に均等に配置した格子点に対し処理を適用しており、今後はその結果を 3D のオブジェクトにマッピングするという形で応用することを検討している。なお本研究では CG 制作ソフト Houdini を用いて提案手法を実装した。

3.1 スケッチ入力と格子点配置

ユーザはまず、2 次元平面上にペイント操作を通して氷や霜の広がりの範囲を入力する。次にユーザによって指定された領域内部に均等にシミュレーション空間を離散化するためのポイントを配置する。本手法では Reiter の手

法 [2] をベースにシミュレーションを行うため、同様にポイントを正六角形状になるよう配置する (図 2 参照)。このとき、スケッチ入力の領域からひと回り大きくポイントを配置する。この外周のポイントは境界条件を計算する際に使用する。ユーザは次に、領域内に配置したポイントから始点となる点 (seed point) を選択する。その際、始点となる点は複数点指定しても問題なくシミュレーションを行うことができる。

3.2 拡散シミュレーション

上述のように設定した領域においてシミュレーションを行う方法について述べる。蒸気場 $u_t(\mathbf{p})$ を氷の状態を表す値として定義する。ここで、 t はフレーム番号、 \mathbf{p} は格子点の位置を表す。まず初期フレームにおいて、蒸気場の各格子点にユーザ指定の初期値 β を割り当てる： $u_0(\mathbf{p}) = \beta$ 。次に、各フレームにおける蒸気場の状態に応じて各格子点を ice 格子とその隣接点、それ以外の点の 3 つの状態に分類する (図 2)。 $1.0 \leq u_t(\mathbf{p})$ の格子点は ice 格子として分類され、この ice 格子が氷・霜として可視化される。各フレームでの計算の過程において、各 ice 格子とその隣接点には定数値 γ を加算する。一方、ice でもその隣接点でもない格子点では、蒸気場に対して拡散方程式を計算する。この拡散方程式の計算において、ice とその隣接点の格子点については値を 0.0 として計算を行う (詳細については [2] を参照いただきたい)。最後にこれら 2 つの結果を組み合わせ、その結果を使用して次のフレームでの蒸気場を計算する (図 2)。

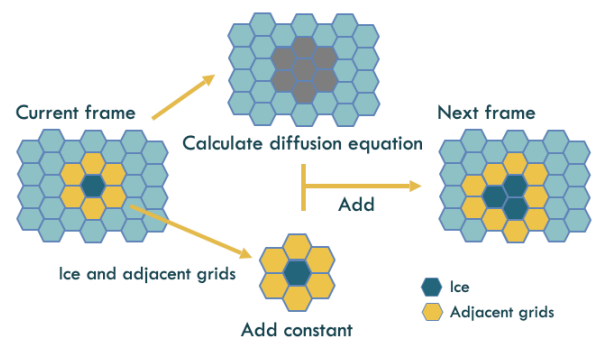


図 2 小領域での Reiter の手法による結晶成長アルゴリズム。

しかし、Reiter の手法では霜の成長のランダム性を表すことはできない。我々はこの問題を解決するために擬似乱数を導入することにより、Reiter の手法を拡張する。この手法では、ice の隣接点である各ポイント (図 2 黄色の格子) に擬似乱数 ($0.0 \leq r_t(\mathbf{p}) \leq 1.0$) を割り当てる。また、ユーザが閾値 r_{th} を設定する。そして、定数値 γ を r_{th} よりも大きい乱数値を持つ隣接点にのみ加算する (式 (1))。ここで式 (1) の $r_t(\mathbf{p}) \leq r_{th}$ の場合は、Reiter の手法による ice でもその隣接点でもない格子点に適用される拡散方程

式である。αはユーザ指定の重み付けの係数で、広がり形状に影響を与える。nn(p)は隣接ポイントセットを表す。

$$u_{t+1}(\mathbf{p}) = \begin{cases} u_t(\mathbf{p}) + D_t(\mathbf{p}) & (r_t(\mathbf{p}) \leq r_{th}) \\ u_t(\mathbf{p}) + \gamma & (r_t(\mathbf{p}) > r_{th}) \end{cases} \quad (1)$$

$$D_t(\mathbf{p}) = \frac{\alpha}{12}(-6u_t(\mathbf{p}) + \sum_{\mathbf{q} \in nn(\mathbf{p})} u_t(\mathbf{q}))$$

閾値の値を大きく設定するほど、複雑な成長の表現が可能になる。この拡張方法により Reiter の手法と比べ、氷のさまざまな成長パターンを表すことができる。

3.3 境界条件

ユーザが行なったスケッチ入力に対し、ひと回り大きく格子点を配置することによって、領域の外周のポイントを境界としてシミュレーションを行う。境界のポイントはice格子として設定されることはなく、境界ポイントの $u_t(\mathbf{p})$ には隣接する6点のうち境界ポイントでないポイントの $u_t(\mathbf{p})$ の平均をとった値を与えて計算を行う。

3.4 初期値の最適化

シミュレーションの初期値βを最適化することで、氷の成長速度を制御する。本稿では最適化の方法として最急降下法を用いる。シミュレーションにおいて各ポイントに与える初期値βが変わると、各ポイントの持つ蒸気場の $u_t(\mathbf{p})$ のシミュレーション過程の値、ひいては広がりの速度が変化する。この初期値βの最適化を行うことで、ユーザの所望のフレームで氷がスケッチした領域の端に到達するよう制御する。

これを実現するために本稿では最適化の目的関数を以下のように定義する。

$$f(\beta) = (T - T_{sim}(\beta))^2 \quad (2)$$

Tは目標終了フレームを表し、ユーザが直接指定する。T_{sim}(β)は初期値βでシミュレーションを行った際の終了フレームを表す。それらの差の二乗が最小になるような初期値βを求める。ここで、終了フレームはiceとなったポイント数が全体のポイント数の9割以上となるフレームとして実験的に設定した。そして、最急降下法の各ステップにおける更新式は以下ようになる。

$$\beta^{i+1} = \beta^i - \eta \frac{\partial f(\beta^i)}{\partial \beta} \quad (3)$$

ここで、iは最急降下法のステップ数であり、ηはユーザ指定の係数である。そして、目標終了フレームとシミュレーション結果の終了フレームの差が十分小さくなった場合に最適化計算を終了し、その時点のβⁱを最終的な初期値として採用する。本稿では、誤差がアニメーション再生時の時間において1/6秒以下であれば十分に小さいと仮定し、

シミュレーションにおいて設定した1フレーム当たりの時間幅から1/6秒に相当するフレーム以内となった場合に収束したとみなす。

4. 実験結果

乱数を導入した氷・霜の広がりのシミュレーション結果を図5に示す。図5では、始点を1点指定した場合の結果を示している。β = 0.35, γ = 0.005のとき、ランダム性を表現する乱数の閾値r_{th}が0.0, 0.4, 0.8のそれぞれの場合について、20フレーム、40フレーム、80フレームの結果を示す。閾値が0.0の場合には従来手法で表現されている均等な氷の結晶成長の様子が表現できている。閾値を大きくするにつれ、閾値0.0の成長と比較して広がりの方向が一様でないランダムな成長を表現することが可能となったことが見て取れる。

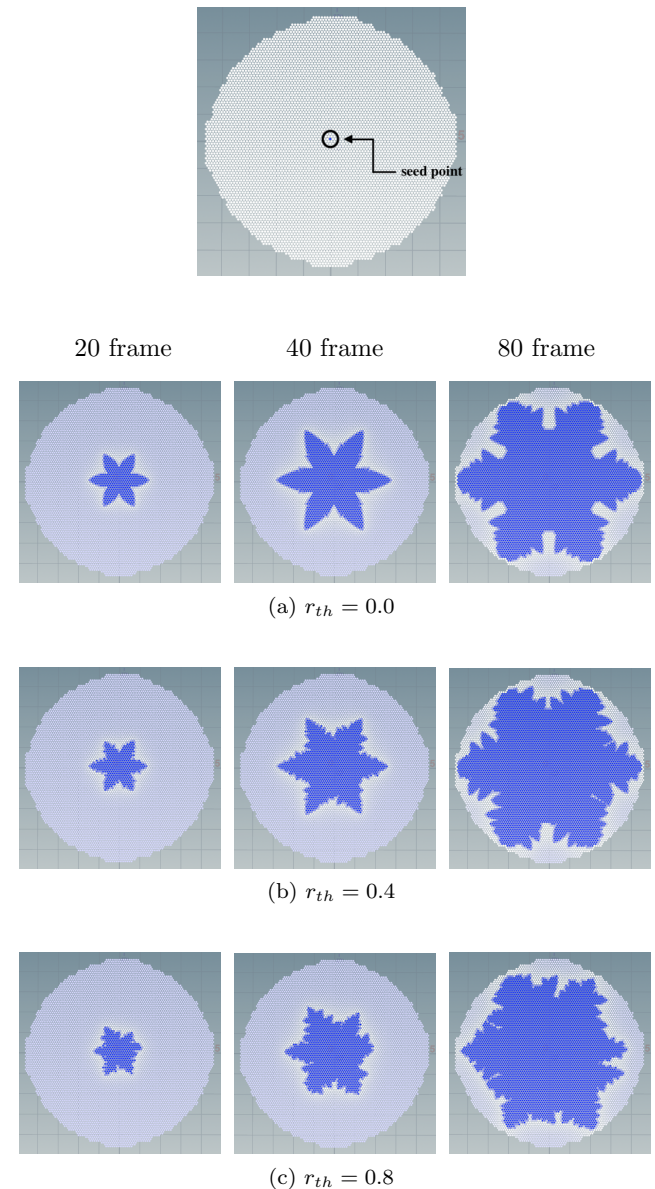


図3 各フレームでの乱数の閾値ごとのシミュレーションの様子。

次に最適化の実験結果を示す。乱数 r_{th} の閾値が 0.0 の場合に目標終了フレームを 130, ステップ幅 η を 1.0×10^{-6} として最急降下法を適用した結果が表 1 である。また、乱数 r_{th} の閾値が 0.8 の場合に目標終了フレームを 130, ステップ幅 η を 1.0×10^{-6} として最急降下法を適用した結果を表 2 に示す。

表 1 閾値 0.0 の最急降下法による初期値

ステップ	初期値	終了フレーム
0	0.25	160
1	0.261856	152
2	0.295357	139
3	0.307597	126

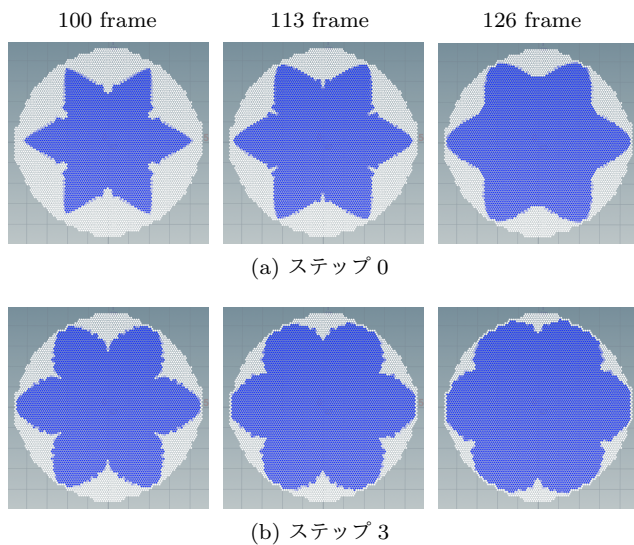


図 4 表 1 ステップ 0, ステップ 3 の各フレームでのシミュレーションの様子。

本実験では作成される映像が 24fps となるようにシミュレーションの時間幅を設定しており、この場合最適化の終了条件は目標終了フレーム ± 4 フレームの間となる。閾値 0.0 の場合、0.8 の場合共に最急降下法により終了フレームが目標終了フレーム ± 4 フレームの間となるような初期値を探索することができていることを確認できた。

5. まとめと今後の課題

本報告では、氷・霜の広がり表現する映像の制作にあたって、スケッチ入力とシミュレーションを用いてインタラクティブに映像を作成するツールに下記の 2 点を導入した。一つ目に、平面のオブジェクト上において、従来手法にランダム性の追加を行った。二つ目に、最急降下法を用いて初期値の最適化を行い、ユーザの所望のフレームでアニメーションの終了するような初期値を決定するシミュレーションを実現した。

我々の実験では、大域的な β の変化と終了フレームの間にはおよそ比例するような関係があるが、 β がわずかに

表 2 閾値 0.8 の最急降下法による初期値

ステップ	初期値	終了フレーム
0	0.25	162
1	0.294132	135
2	0.316651	124
3	0.315713	127

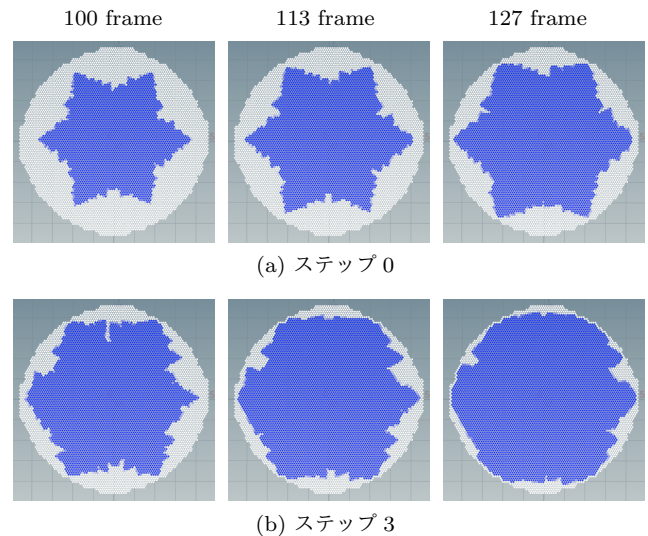


図 5 表 2 ステップ 0, ステップ 3 の各フレームでのシミュレーションの様子。

変化する場合そのような関係とならない場合があることを発見した。特に終了フレームを全ポイントが ice となった時と定義した場合このような問題が発生する。その場合、最急降下法がうまく収束しない場合があった。本稿ではこれに対処するために、実験的に終了フレームを ice となったポイント数が全体のポイント数の 9 割以上となるフレームとした。しかしこの対処法では、意図通りの形状の氷が得られる前に終了フレームとなってしまふ場合がある。そのため、根本的にこの問題を解決するためには最適化の手法を検討する必要があると我々は考えており、今後は最急降下法以外の非線形最適化法を用いて実験を行う予定である。

本稿では初期値 β をシミュレーション空間全体で一つとしたが、これを始点からの各方向毎に変えられるよう定義し、どの方向の領域の境界へも氷・霜の広がりの到達時間が等しくなるよう、広がりの始点から境界までの距離を考慮した最適化を行う予定である。また、平面のオブジェクトにとどまらず、3D のオブジェクトに対しても適用可能とすることが今後の課題である。

参考文献

- [1] Theodore Won-Hyung Kim: *Physically-based simulation of ice formation*, University of North Carolina at Chapel Hill, 2006, Ph.D. thesis.
- [2] Clifford A. Reiter: *A local cellular model for snow crystal growth*, Chaos, Solitons & Fractals, vol.23, 2005, pp.

1111-1119.

- [3] Bo Ren, Jiahui Huang and Ming C. Lin and Shi-Min Hu: *Controllable Dendritic Crystal Simulation Using Orientation Field*, Computer Graphics Forum, vol.37, No.2, 2018, pp. 485-495
- [4] Mikiko Amano and Takayuki Itoh: *Visual Simulation of Ice and Frost with Sketch Input*, ACM SIGGRAPH 2019, Posters, 2019.
- [5] D. Lipton, K Museth and B Sutherland, DW Animation: *Jack's Frost: Controllable Magic Frost Simulations for 'Rise of the Guardians'*, ACM SIGGRAPH 2013 Talks.