

Title	表面の成長による樹木の形状生成
Author(s)	溝口, 敦士; 宮田, 一乗
Citation	芸術科学会論文誌, 13(1): 45-58
Issue Date	2014-03-20
Type	Journal Article
Text version	publisher
URL	<a href="http://hdl.handle.net/10119/12898">http://hdl.handle.net/10119/12898</a>
Rights	溝口敦士, 宮田一乗, 芸術科学会論文誌, 13(1), 2014, 45-58. 本著作物は芸術科学会の許可のもとに掲載するものです。
Description	

## 表面の成長による樹木の形状生成

溝口敦士（非会員）      宮田一乗（正会員）

北陸先端科学技術大学院大学 知識科学研究科

## Modeling Trees Using Surface Growth Simulation

Atsushi Mizoguchi(Nonmember)      Kazunori Miyata(Member)

School of Knowledge Science, Japan Advanced Institute of Science and Technology

{mizoguchi, miyata} @ jaist.ac.jp

### アブストラクト

本研究では、表面の成長による樹木の形状生成を行った。樹木は様々な表面形状の特徴を持っており、樹木形状を生成するにはこの表面形状は重要な要素である。樹木の表面形状を生成するために、樹木の成長をシミュレートする。樹木は細胞分裂によって成長する。この細胞分裂による成長をポリゴンの変形を用いて近似する。ポリゴンメッシュの各頂点は枝を伸ばす頂端分裂組織、枝を太らせる形成層、樹皮下にあり樹木の構造を支えている木部の役割を演じる。ポリゴンの各頂点はこの役割によって移動し樹木形状を作る。この時、頂点の移動によって頂点がない領域が発生する。本手法では頂点がない領域では成長をシミュレートできない。そこで、樹木全体に頂点を配置するために、ポリゴンメッシュの最適化を行う。頂端分裂組織の成長はパラメトリック L-system で定義し、形成層の成長と木部の腐敗は複数のパラメータで定義する。また、本研究では特にコブ、樹洞、障害物と衝突した際の反応をシミュレートした。これにより様々な表面形状を持った樹木形状が生成可能となった。

### Abstract

In this study, we propose a method for simulating tree growth that considers several features of a tree's surface. The phenomena of cell division are considered for simulating tree growth. To simulate tree's growth, a tree is modeled as a polygonal mesh that grows by displacing the vertices of the polygonal mesh. Changing speed of the displacing, several tree's surfaces are generated. In this study, we focus on knots, hollows, and extra wood caused when tree growth encounters an immovable object. Knots are generated by unusual and uneven growth of the cambium cells; a hollow is caused by rotten unsound xylem; and extra wood is made to protect the body. To define a tree shape, branching structures of trees are controlled by parametric L-system, and surface shapes are specified by some parameters. Subsequently, we generate tree models with several surfaces, including burls, hollows, and extra wood.

## 1. はじめに

植物は山から街中まで地球上のいたる場所で生育している。そのため、コンピュータグラフィックスで作成された映画やビデオゲームといったメディア内においても、膨大な数の植物が登場する。コンピュータグラフィックスで植物を表示させるためには、その植物の形状データが必要である。植物の形状は非常に複雑で数も膨大であるため、人の手で形状データを作成することは非常に手間のかかる作業である。そのため、植物の形状データの作製を支援し、コンピュータグラフィックスの制作コストを低減する手法が求められている。樹木形状の作成を支援する手法は数多く提案されており、それらの手法では計算によって樹木形状を生成している。

樹木はその表面に樹洞やコブといった様々な凹凸形状を持っている。そのため、現実味のある樹木形状を生成するためには表面の凹凸形状を考慮する必要がある。しかしながら、樹木形状の生成を行っている既存手法では、主に樹木の枝ぶりの生成に焦点が当てられている。そのため、樹木表面の凹凸形状は考慮されておらず、生成される樹木形状の表面は凹凸を持たない。そこで、本研究では現実の樹木のような、表面に凹凸形状を持つ樹木形状生成を目的とする。本研究で扱う樹木の表面形状はコブ、樹洞、障害物との衝突による変形である。

本手法では凹凸を持った樹木形状を生成するために、現実の樹木の成長を模倣する。樹木は細胞分裂によって成長する。本手法では、細胞分裂による成長をポリゴンメッシュの変形でモデル化する。初期条件として与えられた閉じたポリゴンメッシュを、植物の成長に沿った形で変形させることで成長をシミュレートする。成長のシミュレーションは、ポリゴンメッシュを構成する頂点の移動とポリゴンサイズを均一に保つ最適化により行う。

本論文で提案するシミュレーションモデルによって、コブや樹洞、障害物との衝突による変形した樹木形状などがパラメータの指定のみで生成可能になった。本手法は、森林のような数が膨大で詳細な表面形状を持った樹木をモデリングする必要がある場面にて有用であると考えられる。

## 2. 関連研究

樹木の形状を生成する手法は数多くあり、プロシージャルモデリング、インバースモデリング、計測データからの再構築、スケッチベースモデリングといった手法が提案されている。また、表面形状を生成する手法も提案されている。

### ・プロシージャルモデリング

Ulam[1]やHonda[2]らが樹木の枝ぶりを生成する手法を提案した事を始めとし、様々な枝ぶりを生成する手法がこれまでに提案されている。Prusinkiewiczらが、Lindenmayerが提唱したL-system[3]を用いて樹木の枝ぶり生成を行う手法[4]を提案した。

L-systemはその後様々な発展した[5]。例えば、Stochastic L-system[4]では確率の要素を組み込むことで、枝ぶりなどにばらつきを与え、自然な植物形状を生成している。パラメトリッ

クL-system[6][7]ではステップ毎に変数を変化させ、その値によって条件分岐などを行っている。樹木は成長するときに環境から影響を受けるが、環境からの影響を組み込んだOpen L-systemが提案され、Prusinkiewiczらは枝の伸長を制御し、生け垣などの刈り込んだ樹木形状を生成する手法[8]を、Mêchらは樹木と環境の相互関係をL-Systemに組み込んだ手法[9]を提案した。その他にも、Palubickiらは一つの樹木内で起こる相互関係による自己組織化をL-Systemに組み込んだ手法[10]を提案した。

本研究ではユーザーが成長シミュレーションのパラメータを与えるためにパラメトリックL-systemを用いた。

### ・インバースモデリング

プロシージャルモデリングの煩雑なパラメータ調整を軽減させるためにインバースモデリングが提案されている。Pirkらは既存の樹木形状を環境情報によって変化させる手法[11]や、既存の樹木形状の過去の形状を推測する手法[12]を提案した。これらの手法によって、一つの樹木形状から、煩雑なパラメータ調整を必要とせず、様々な樹木形状が生成可能になった。

### ・計測データから再構成

Cheng[13]らやTan[14]らは一枚の写真から樹木形状を推測し、生成する手法を提案し、Shlyakhter[15]、Neubertら[16]、Tanら[17]は複数の写真から枝ぶりを推測する手法は提案した。Linvy[18][19]らやXu[20]らはレーザースキャンで樹木形状を計測し、取得したポイントクラウドから樹木形状を再構成する手法を提案した。

これらの手法では、プロシージャルモデリングのように煩雑なパラメータ調整が必要ない。その一方で実際にある樹木を測定するためには高価な機材と、煩雑な測定作業を必要とする。また、実際に存在する樹木形状しか生成できないという欠点もある。

### ・スケッチベースモデリング

直観的な樹木のモデリングを行うために、2Dのスケッチによるモデリング手法が数多く提案されている。Okabeらは幹のスケッチと枝の規則を入力として樹木形状を生成し[21]、Ijiriらは幹や枝、葉の形状に至るまでスケッチによって定義する手法[22]を提案した。また、Longayらはスケッチを用いてインタラクティブに樹木形状を生成する手法[23]を提案した。これらスケッチベースの手法では、こぶやうろなどの複雑な表面形状を持つ樹木のモデリングは行い難い。

手作業で3Dメッシュの表面に凹凸をデザインするツールとしてZBrushが知られる。このツールを用いると、3Dメッシュの表面を彫刻するような編集が可能であり、高品質な表面形状をデザインできる。しかしながら、森林のモデリングなど、膨大な数の樹木をモデリングする必要がある場面でこのツールを用いるのは現実的ではない。提案法では、様々な表面形状を持つ樹木モデルが成長シミュレーションにより自動生成されるため、膨大な数の樹木が必要な場面で提案法はとりわけ役に立つ。

### ・成長シミュレーションによる生物の表面形状生成法

樹木形状を生成している既存研究では、樹木の枝ぶりを線分によって表現している。そして、その線分に沿って円柱、Generalized Cylinder[24]、陰関数曲面などを用いて表面の形状を

定義している。しかしながら、このような手法で定義された表面形状は均一で凹凸を持たず、現実の樹木に存在するコブや樹洞などの凹凸のある表面形状を個別に生成することはできない。

ポリゴン上に凹凸を生成する手法としては、ポリゴンで表現された弾性体のポリゴンの辺の長さをコントロールしてひだ状の凹凸をつける手法[25]や、この手法を発展させ、大規模変形を含んだ様々な形状が生成可能な手法[26]などが挙げられる。これらの手法は汎用性が高く、様々な表面形状の生成に有用であると考えられる。また、これらに似た手法として花びら[27]や落ち葉[28]を弾性体として考え形状の変形をシミュレートする研究が行われている。ポリゴン表面に凹凸を生成する手法[25][26]は汎用的ではあるが、樹木の形状生成には適さないと考える。すなわち、これらの既存手法ではポリゴンに対して水平方向に加わる力を基に垂直成分の変形を計算しているため、細胞分裂による樹木の成長との乖離がある。さらに、既存手法ではポリゴンの垂直方向に対してのみ凹凸を生成するため、枝の先端のように尖った形状が生成できない欠点がある。

本手法では実際の樹木の成長モデルを参考にし、様々な特徴を持った表面形状を自動生成する。これにより、膨大な数の樹木が必要な場面で役立つと考える。

本手法でも用いているポリゴンの変形による形状生成手法として、ポリゴンメッシュの変形によってサンゴの形状を生成する手法[30]がある。このサンゴの形状を生成する手法ではサンゴをポリゴンメッシュで近似し、ポリゴンメッシュを構成する各頂点は、サンゴを形成するポリプとして振る舞う。ここで頂点が移動する方向や量はアトラクタ(水中の餌)で決定される。アトラクタは空間に配置され、このアトラクタに向かって成長することでサンゴの成長をシミュレートしている。

本研究の頂点の移動や最適化といった基本的な操作は既存手法[30]とほぼ共通である。そして本研究の特徴は、対象である樹木の成長特性に合わせて、後述のコブ球やパラメトリックL-systemなどを形状の指標として扱っている点にある。本手法により、従来の樹木形状の生成手法では扱っていない、詳細な表面形状を持った樹木形状の生成が可能になった。

### 3. シミュレーション概要

本手法では樹木表面の成長をシミュレートすることによって、樹木形状を生成する。樹木は図1のように枝を伸ばす頂端分裂組織と枝を太らせる形成層によって成長する[29]。この2種類の細胞は樹木表面に存在し、樹木形状を決定している。そのため、樹木表面の成長をシミュレートすることで樹木全体の成長をシミュレートし、樹木形状を生成できる。

本手法では、樹木表面はポリゴンメッシュで表現する。このポリゴンメッシュは頂端分裂組織、形成層、腐敗木部といった実際の樹木の成長に沿った変形をし、樹木形状を生成する(図2)。

図3に樹木の成長シミュレーションのフロー図を示す。成長のシミュレーションは、頂点の移動とポリゴンの最適化の2つの要素から構成される。

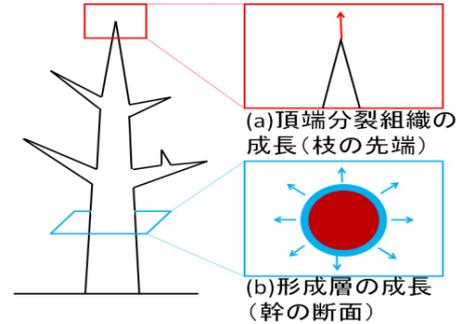


図 1 樹木の成長モデル

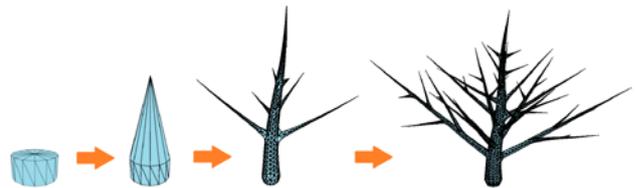


図 2 ポリゴンの変形による成長シミュレーション

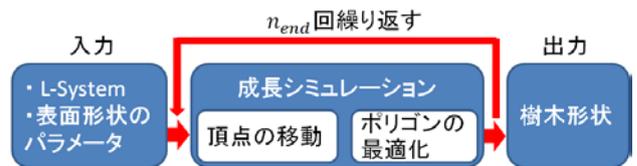


図 3 樹木の成長シミュレーションフロー図

## 4. ポリゴンメッシュによる成長シミュレーション

本手法では樹木形状は閉じたポリゴンメッシュで定義され、このポリゴンメッシュを変形することで、樹木の形状を生成する。ポリゴンメッシュの各頂点は頂端分裂組織、形成層、腐敗木部として振る舞う。この変形に関する様々なパラメータはパラメトリックL-systemの変数とユーザー指定の値によって決定される。

ポリゴンの変形を行うと、ポリゴンの大きさが不均一になる。ここでの均一なポリゴンとは各ポリゴンの辺の長さが指定された範囲内にある状態とする。ポリゴンの大きさが不均一になると、頂点が無い部分が生じ、樹木表面全体を均一にシミュレーションできない。そこで、ポリゴンの大きさを均一に保つための最適化を行う。

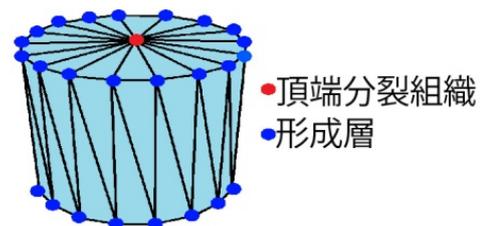


図 4 初期形状

### 4.1 初期形状

樹木形状の初期形状は閉じたポリゴンメッシュである。閉じ

たポリゴンメッシュであればどのような形状でも良く、本論文では図4のような、直観的に初期形状と分かる、樹木の種の形に近い多角柱を用いた。本手法ではこの閉じたポリゴンメッシュを変形することで、成長をシミュレートする(図2)。

### 4.2 ポリゴンの成長

図5に示すように、ポリゴンの頂点 $P_i^n$ はそれぞれ成長ベクトル $v_i^n$ と座標値 $p_i^n$ を持ち、シミュレーションステップごとに移動する(式(1))。

$$p_i^{n+1} = p_i^n + v_i^n \quad (1)$$

$n$ は現在のステップ数を表し、ステップ毎に座標値に成長ベクトルを加えることで成長をシミュレートする。

成長ベクトルは樹木の性質によって決定される。頂端分裂組織頂点では枝を伸ばすように、形成層頂点であれば枝を太くするように、腐敗木部頂点であれば幹に穴をあけるように成長ベクトルを決定する。すなわち、細胞分裂や腐敗による樹木の形状変形を、1ステップあたりに成長する量と方向として表現したものが成長ベクトルである。成長ベクトルの計算手法は5.1で後述する。

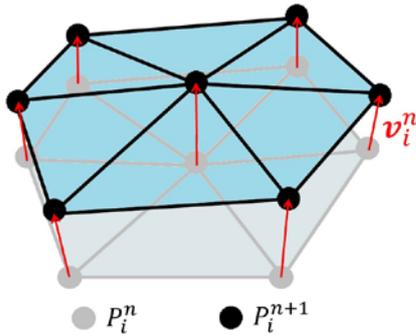


図5 ポリゴンの成長

### 4.3 ポリゴンメッシュの最適化

頂点の移動によって、ポリゴンメッシュには粗密が生じる。樹木表面全体を均等にシミュレートするために、ポリゴンメッシュの最適化を行う。最適化にはポリゴンメッシュを均一に保つための、ポリゴンの分割処理と頂点の結合処理の2つがある。

図6にポリゴンの分割処理を示す。ポリゴンの辺の長さがユーザー設定の閾値 $T_s$ より長くなった時、ポリゴンを分割する。ポリゴン分割によって新しく作成された頂点 $P_{new}$ の座標 $p_{new}$ は頂点 $P_i$ と $P_j$ の中点座標とする。 $P_{new}$ のタイプは $P_i$ と $P_j$ のタイプによって決定する(表1)。ここで、木部が露出した部分以外、樹木表面は形成層でおおわれている。そのため木部を含まない辺が分割される場合は、新しい頂点を形成層とする。

図7に頂点の結合処理を示す。辺を構成する頂点間の距離がユーザー設定の閾値 $T_m$ より短くなった時、その頂点同士を結合し、新しい頂点を作る。 $T_m$ は $T_s$ の半分以下の値を用いる。これは $T_m$ が $T_s$ の半分より大きな値をとると、ポリゴン分割と頂点結合の2つの処理が最適化の対象となり、最適化後のポリゴン形状を決定できないためである。頂点の結合処理によって生じた $P_{new}$ のタイプと座標 $p_{new}$ は $P_i$ と $P_j$ の種類によって決定する(表2)。ここで、頂端分裂組織頂点同士では結合処理を行わない。これは頂端分裂組織は枝ぶりを決定しているため、頂端分裂組

織同士を結合すると、パラメトリック L-system で指定した枝ぶりを生成できなくなるためである。同様に頂端分裂組織頂点と他種の頂点が結合する場合も新しい頂点は頂端分裂組織頂点とし、枝ぶりを保護する。また、頂点の結合処理は前ステップより頂点間の距離が短くなった場合に限る(式(2))。これは枝の先端では頂点間の距離が短くなる傾向があるため、この条件式がなければ枝先端の形状が破壊されるためである。

$$|p_i^n - p_j^n| < |p_i^{n-1} - p_j^{n-1}| \quad (2)$$

以上のポリゴンの最適化処理により、ポリゴンのサイズが均一に保たれ、樹木表面全体の成長を均一にシミュレートできる(図8)。

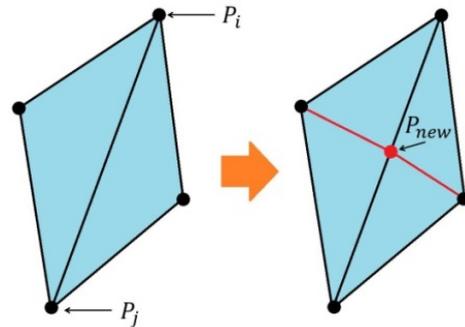


図6 ポリゴン分割処理

表1 ポリゴン分割時の頂点タイプ  
(AM:頂端分裂組織 CA:形成層 XY:木部)

$P_i$ のタイプ	$P_j$ のタイプ	$P_{new}$ のタイプ
AM	AM	CA
AM	CA	CA
AM	XY	CA
CA	CA	CA
CA	XY	XY
XY	XY	XY

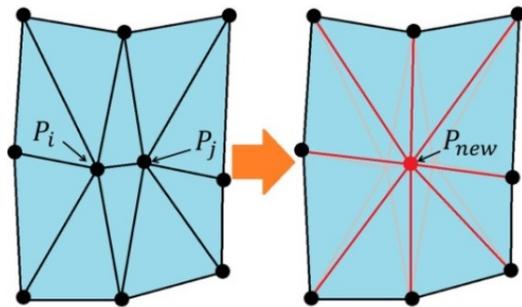


図7 頂点結合処理

表2 頂点結合時の頂点タイプと位置  
(AM:頂端分裂組織 CA:形成層 XY:木部)

$P_i$ のタイプ	$P_j$ のタイプ	$P_{new}$ のタイプ	$p_{new}$
AM	AM	-	-
AM	CA	AM	$p_i$
AM	XY	AM	$p_j$
CA	CA	CA	$(p_i + p_j)/2$
CA	XY	CA	$p_i$

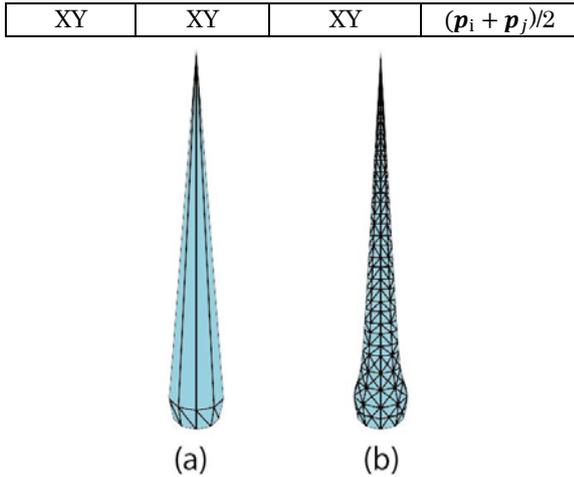


図 8 ポリゴンメッシュの最適化(a)なし(b)あり

## 5. 樹木形状の生成

本章では4章で述べたポリゴンメッシュの変形を用いて、樹木形状を生成する手法を述べる。

### 5.1 成長ベクトル

ポリゴンメッシュを構成する各頂点は、樹木表面の性質に依存して変形する。それぞれの頂点は頂端分裂組織、形成層、または腐敗木部として振る舞う。本節では頂端分裂組織と形成層の成長ベクトルの計算方法を示す。腐敗木部の成長ベクトルは7.1で述べる。

式(3)に頂端分裂組織である頂点の成長ベクトルを示す。

$$v_i^n = s_{ai}^n d_i^n \quad (3)$$

$s_{ai}^n$ ,  $d_i^n$ はそれぞれ、 $n$ ステップ目における成長速度、成長方向を表すベクトルである。頂端分裂組織は枝ぶりを決定する働きを持つ。成長速度と成長方向は従来研究と同様に、パラメトリック L-System の変数によって値を決定する。これによりパラメトリック L-System で記述した曲線に沿った枝が生成される (図9)。

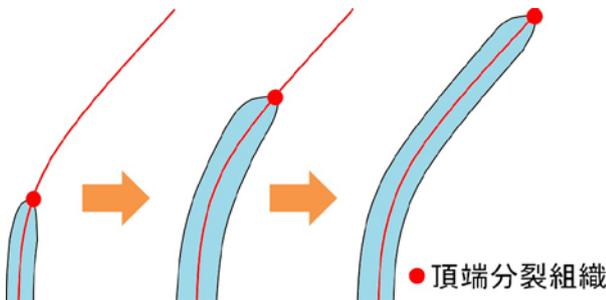


図 9 曲線に沿った頂端分裂組織の成長

式(4)に形成層ベクトルの成長速度を示す。

$$\begin{aligned} v_i^n &= S_{ci} d_i^n \\ d_i^n &= N_i^n \end{aligned} \quad (4)$$

$S_{ci}$ は形成層頂点の成長速度である。ポリゴン分割によって新しく生成された形成層頂点の成長速度 $S_{ci}$ には、頂端分裂組織が持つパラメトリック L-System の形成層の成長速度に関する変数の値を用いる。これにより、枝ごとの太さの差異を表現可

能になる。形成層は枝を太くする働きを持つ。そこで、枝の肥大成長をシミュレートするために、成長方向 $d_i^n$ は頂点法線方向 $N_i^n$ とする。形成層頂点を頂点法線方向に成長させることで、枝の肥大成長をシミュレートする。頂点法線はシミュレーションステップ毎に面法線から式5を用いて計算する。

$$N_i^n = \frac{\sum_{l \in C} M_l^n}{|\sum_{l \in C} M_l^n|} \quad (5)$$

$M_l^n$ はポリゴンの面法線であり、 $C$ は頂点を含むポリゴンの集合である。法線を計算する頂点を含むポリゴンの面法線を足し合わせ、足し合わせた法線を正規化することで、頂点法線を計算する。

### 5.2 枝の生成

樹木は数多くの枝を持っている。これをポリゴンメッシュの変形による成長シミュレーションで実現するために、以下のようなステップで枝を生成する(図10)。

- (a)枝となる頂点を形成層頂点から選び、この頂点を芽頂点 $p_{bud}$ とする。芽を作る頂端分裂組織頂点と近傍の複数の形成層頂点から、形成層頂点の法線 $N_i^n$ と芽の成長方向 $d_{bud}$ のなす角が一番小さい形成層頂点を芽頂点として選ぶ。芽の成長方向 $d_{bud}$ はパラメトリック L-system を通してユーザーが指定する。
- (b)芽頂点は発芽するまでは形成層頂点と同様に振る舞う。発芽するまでのシミュレーションステップはパラメトリック L-system で指定する。
- (c)芽頂点が発芽し、頂端分裂組織頂点として振る舞う。
- (d)頂端分裂組織頂点が成長することで新しい枝が生成される。

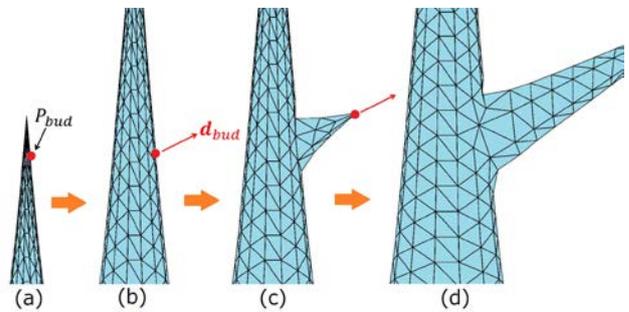


図 10 枝の生成

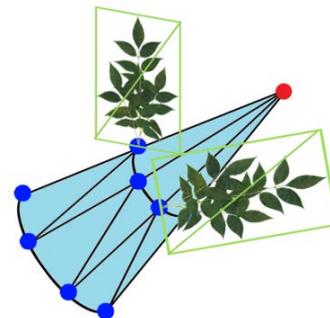


図 11 板ポリゴンによる細かい枝と葉の表現

### 5.3 その他のシミュレーション

本手法では細かい枝を生成することは困難である。これはポ

リゴンで枝を表現するとき、枝の根本の最小半径は頂点結合の閾値 $T_m$ であることに起因する。そこで、枝の根本の半径が $T_m$ より小さいような細かい枝や葉は、板ポリゴンに事前に用意された単一のテクスチャを張り付けることで実現する(図 11)。この板ポリゴンを頂端分裂組織に近い形成層頂点にランダムに配置することで、自然な細かい枝と葉を生成する。

樹木表面は、成長の過程で自身の樹木表面に衝突することがある。この時、衝突した樹木表面同士は融合する[29]。これをシミュレートするために、成長する頂点と樹木表面を構成するポリゴンの衝突判定を行う。成長する頂点がポリゴンと衝突した場合、成長する頂点をポリゴンの頂点と結合する。結合するポリゴンの頂点は成長する頂点と一番距離が短い頂点とする。

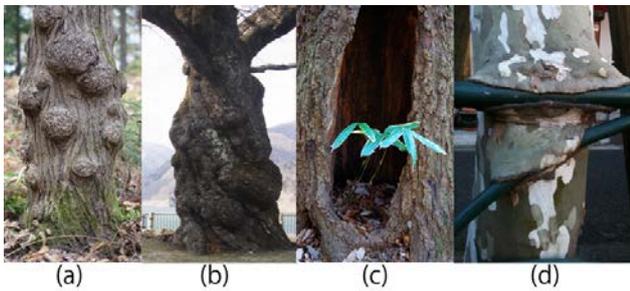


図 12 樹木表面の様々な特徴  
 (a) 異常成長によるコブ (b) 養分の偏りによるコブ  
 (c) 樹洞 (d) 障害物との衝突による変形

シュ上の形成層頂点 $P_{center}$ の座標値 $p_{center}$ とその頂点からの半径  $R$ ,  $R'$ で定義される球である。この球の範囲内にあるオレンジ色と赤色の形成層頂点の移動速度を速くすることで球状のコブを形成する。コブの形成には、関数 $K_1(p_i^n, p_{center}, R, R', k)$ を図 14 のように定義する。コブの中心頂点付近、すなわち赤い球内にあるコブの成長係数を一定値にすることで、円錐のようなコブの形成を防ぎ、球状のコブの形成が可能となる。

### 6.3 養分の偏りによるコブの生成

樹木表面の成長速度は、師管を流れる養分量に依存している。養分量は枝の伐採などにより、不均一になることがある。このように養分に偏りができると、図 12(b)のような師管の流れに沿ったコブが生じる。

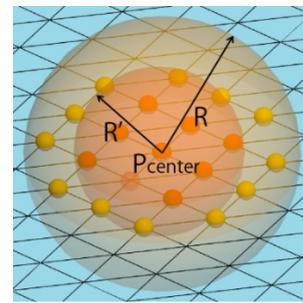


図 13 コブ球

## 6. コブのシミュレーション

樹木は様々な原因により、その表面にコブを作る。樹木は樹皮下の虫や細菌により異常成長し球状のコブを作り、師管を流れる養分の不均一によって師管の流れに沿ったコブを作る[29]。

### 6.1 コブを考慮した成長ベクトル

コブの成長をシミュレートするために、形成層頂点の成長ベクトルを変化させる(式(6))。

$$v_i^n = \left( 1 + \sum_{i \in D} K_1(p_i^n, p_{center}, R, R', k) \right) S_{c_i} d_i^n \quad (6)$$

$K_1(p_i^n, p_{center}, R, R', k)$ はコブの成長速度を変化させるためのコブの成長係数を求める関数であり、 $D$ はコブ球すべての集合である。

### 6.2 異常成長によるコブの生成

形成層は虫や細菌が樹皮下に入り込むことによって、図 12(a)のような異常成長を起こしコブを形成する。異常成長で作られるコブが球状であることは、観察により確認している。これらの知見から、虫や細菌が入った樹皮下のある一点を中心とした周囲で異常成長が起こることでコブが形成されると考え、モデル化した。

コブの形状を定義するために、コブ領域を示す球を用いる(図 13)。この領域を“コブ球”とする。コブ球とは、ポリゴンメッ

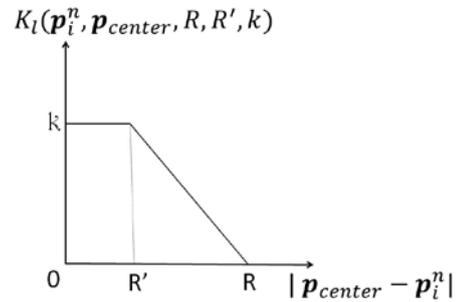


図 14 コブの成長係数の関数

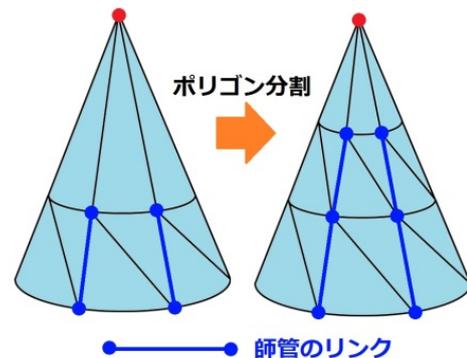


図 15 師管のリンク生成

養分の偏りによるコブの生成を行うために、師管の流れを作成する。図 15 に示すように、師管はポリゴンの辺として表現される。初期状態の円錐状のポリゴンは縦方向に師管のリンクを持っている。このリンクは頂点分割処理のたびに更新される。

頂端分裂組織と形成層頂点の間で頂端分裂が生じたとき、新しくできた頂点は形成層頂点が持っている師管とリンクする。

師管は直線的に成長するだけでなく、枝や幹の周囲を回転するように成長する[31]。これをシミュレートするために、頂端分裂組織頂点と形成層頂点の間でポリゴン分割処理が生じた場合、新しい頂点の位置を頂端分裂組織頂点の成長方向軸周りに回転させる(図16)。回転量 $\theta_i^n$ は式(7)によって計算される。

$$\theta_i^n = \theta_p \sum_{t=0}^n |v_t^i| \quad (7)$$

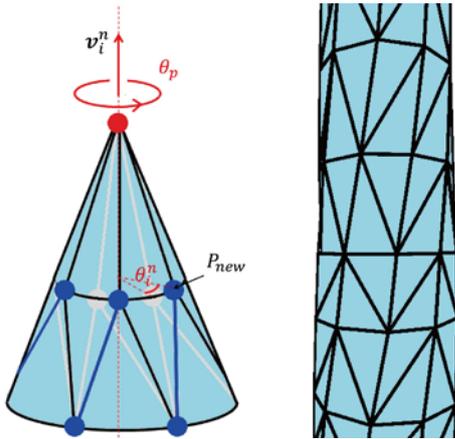


図 16 師管リンクのねじれ

$\theta_i^n$ は頂端分裂組織の移動量に比例し、 $\theta_p$ は移動量あたりの回転量である。ステップ毎の回転量を足し合わせることで、現在の回転量を得ることができる。

以上のように作成された師管上にコブ球を配置する。これにより異常成長によるコブの生成同様に、師管に沿ったコブが生成され、樹木表面に流れる養分の偏りによるコブが生成される。

## 7. 樹洞のシミュレーション

樹木は表面の樹皮が傷つくと、内部の木部が露出し、その木部が腐敗することによって図12(c)のような樹洞を作る[29]。この樹洞の生成をシミュレートするために、頂点を樹木内部に向かって成長させる。これによって樹木表面に樹洞を生成する。

樹洞には2つの成分がある。それは樹木表面に開いた穴(図17(a))とその周囲に現れるコブ(図17(b))である。この2つの特徴をシミュレートすることで、現実味のある樹洞を生成する(図17(c))。

### 7.1 木部の成長ベクトル

樹洞の形を決定するために、腐敗木部を演じる頂点を選択する。腐敗木部頂点の選択には閉じたポリゴンメッシュを用いる(図18)。このポリゴンメッシュ内に存在する頂点を腐敗木部頂点として選択する。この腐敗木部頂点の成長ベクトルを式8のように制御することで、樹洞を生成する。

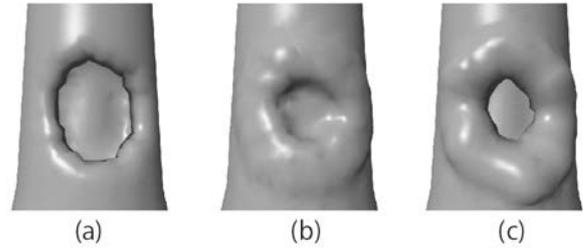


図 17 樹洞の成分(a)穴(b)コブ(c)穴とコブ

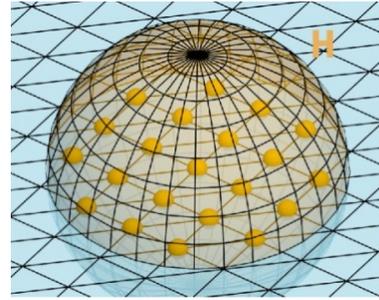


図 18 ポリゴンメッシュによる樹洞範囲の指定

腐敗木部頂点の成長ベクトルは式(8)で計算される。

$$\begin{aligned} v_i^n &= -S_h d_i^n \\ d_i^n &= N_i^n \end{aligned} \quad (8)$$

$S_h$ は木部の腐敗する速度であり、 $S_h$ はユーザー指定の値とする。木部が法線の逆方向に腐敗していくことで、図17(a)のように樹木表面に抉られたような樹洞が生成される。

### 7.2 巻き込み

樹木は傷ついた樹皮を塞ごうとする性質がある。この性質をシミュレートするために樹洞周囲の成長方向ベクトルを変化させる(式(9))。

$$\begin{aligned} d_i^n &= N_i^n + A W_i^n \\ W_i^n &= \frac{\sum_{l \in E} \frac{(p_l^n - q_l^n)}{|(p_l^n - q_l^n)|}}{\left| \sum_{l \in E} \frac{(p_l^n - q_l^n)}{|(p_l^n - q_l^n)|} \right|} \end{aligned} \quad (9)$$

$p_l^n$ は腐敗木部頂点の位置、 $q_l^n$ はその腐敗木部頂点を持ったポリゴン群 $E$ に含まれる頂点の位置である。 $A$ は巻き込みの量を決定する係数で、ユーザーが指定する。腐敗木部頂点の近傍頂点が、腐敗木部頂点に向かって成長する。これにより、樹洞を塞ぐような成長をシミュレートできる。樹洞をふさぐ速度は定数 $A$ で制御可能である

### 7.3 コブの生成

樹木は傷ついた部分を塞ぐために、傷ついた周囲の形成層の成長を促す特徴を持つ。そのため、樹洞の周りにコブが生じる。樹洞の周囲のコブを生成することによって、現実味のある樹洞形状を生成する。コブを生成するために、今まで述べたコブの生成と同様にコブ球を用いる。近傍頂点に木部頂点を持つ形成層頂点にコブ球を配置することによって、図17(b)のような樹洞周囲にコブを生成することができる。このように穴(図17(a))とコブ(図17(b))を組み合わせることで図17(c)のような現実味のある樹洞形状が生成可能である。

## 8. 障害物との衝突による変形シミュレーション

樹木は金属製のフェンスといった移動しない障害物と衝突すると、図 12(d)のように衝突部周辺の成長が促進される[29]。これは、樹木と障害物の衝突部分に大きな力が加わるためである。樹木は衝突部分にかかる圧力を弱めるために、衝突部分周辺の成長を促進させ、障害物を包み込むように成長する。この特徴をシミュレートするために3つの処理を行う。

①障害物と衝突した頂点は成長を停止し、頂点が障害物の内部に侵入しないようにする。②障害物と衝突した頂点の近傍頂点の移動速度を速くする。③樹木表面が障害物を巻き込むように成長方向を制御する。

以上を考慮することによって、障害物と衝突した時の樹木表面の変形をシミュレートできる。

障害物との接触による成長の促進をシミュレートするために、コブ球を用いる。障害物と接触した頂点にコブ球を配置することによって、接触点付近の成長速度を速くする。

樹木は障害物を覆うように成長する。これをシミュレートするために成長方向ベクトルを制御する(式(10))。

$$d_i^n = N_i^n + BW_i^n$$

$$W_i^n = \frac{\sum_{l \in F} \frac{p_l^n - r_l^n}{|p_l^n - r_l^n|}}{\left| \sum_{l \in F} \frac{p_l^n - r_l^n}{|p_l^n - r_l^n|} \right|} \quad (10)$$

$p_l^n$ は障害物と接触している頂点の位置、 $r_l^n$ はその接触頂点を持ったポリゴン群 $F$ に含まれる頂点の位置である。 $B$ はどの程度巻き込みを行うかの係数であり、ユーザーがその値を指定する。接触している頂点の近傍頂点が、接触している頂点に向かって成長することにより、障害物を覆うような成長がシミュレートできる。

## 9. テクスチャマッピング

樹木をレンダリングする際には、樹木の形状モデルにテクスチャを張り、表面の細かな形状を表現する手法が用いられる。形状データにテクスチャを張るためには、頂点それぞれに2次元のテクスチャ座標を設定する必要がある。そこで、本章では生成した樹木形状にテクスチャ座標を設定する手法について述べる。

テクスチャ座標を計算するために、樹木を枝ごとに分解する。ポリゴンメッシュを枝ごとに分解するために、頂点がどの頂端分裂組織から生じたかの情報を利用する。頂端分裂組織を含むポリゴンの分割処理によって新しい形成層頂点が生じた場合、その生じた形成層頂点は生成元の頂端分裂組織の枝に属するものとする。また、属する枝が違う形成層頂点間でポリゴン分割処理が起こった場合、新しい形成層頂点は先端に近い枝に属するものとする。属する枝が違う形成層頂点同士で頂点結合が起こった場合も、新しい形成層頂点は先端に近い枝に属する。これにより、図19のように樹木を枝ごとにセグメンテーションすることができる。

また、木部と形成層に用いるテクスチャが異なるため、木部と形成層についてもセグメンテーションする。このセグメンテーションはポリゴンが木部頂点を含んでいるかを指標とする。ポリゴンを構成する頂点の一つでも木部頂点があった場合、そのポリゴンは木部に属するものとする。図20は木部と形成層をセグメンテーションした結果である。

セグメンテーションされた各要素は円筒マッピングでテクスチャ座標を計算する。師管にねじれがある場合は、テクスチャマップも同様のねじれの回転量を足し合わせる。このテクスチャ座標を用いて、図21に示すような循環している樹皮のテクスチャをタイリングすることで、自然な樹木表面をレンダリングする。

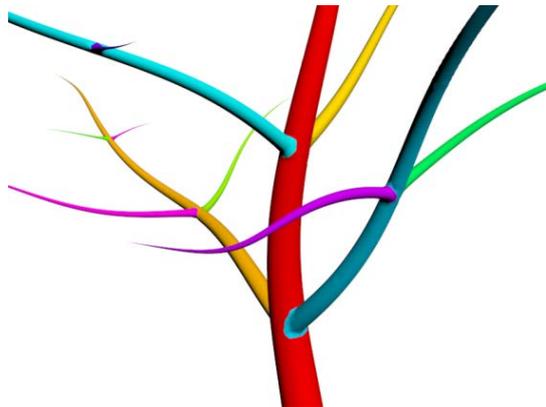


図 19 枝のセグメンテーション結果

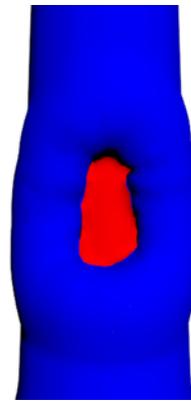


図 20 木部のセグメンテーション  
(青：形成層，赤：木部)

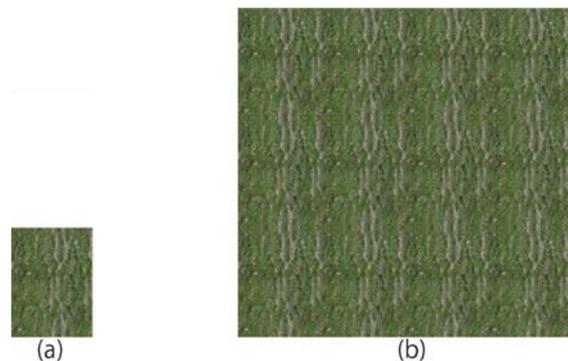


図 21 循環テクスチャ (a)循環テクスチャ  
(b)循環テクスチャをタイリングした結果

## 10. 結果

図 22 は様々な角度での枝の生成結果である。ポリゴンの最適化と、表面の融合によってスムーズに幹と枝が接続されていることが確認できる。

図 23(a)は表面にコブや樹洞がない様々な生成結果である。その表面形状はなめらかであり、既存の研究と同等の表面形状を持つ。

図 23(b)-(d)に異常成長によるコブの様々な生成結果を示す。コブ球によって球状のコブが生成されていることが分かる。異常成長によるコブの形状はコブ球のパラメータによって制御可能である。コブの大きさは  $R$  で、コブの高さは  $k$  によって制御可能である。

図 24(a)-(c)に養分の偏りによるコブの生成結果を示す。養分の偏りによるコブの形状は、異常成長によるコブと同様に、コブ球で制御可能である。ねじれはねじれ速度  $\theta_p$  によって変化させることができる。

図 25(a)-(c)に様々な形状の樹洞を持った樹木の生成結果を示す。結果画像から穴とコブを持った樹洞が生成されていることが分かる。樹洞の形状は樹洞の範囲を表すポリゴンメッシュ、樹洞の深さは成長速度  $S_h$ 、巻き込み係数  $A$ 、周囲にできるコブはコブ球のパラメータによって制御可能である。

図 26(a)-(c)に障害物と衝突した樹木の様々な生成結果を示す。障害物と衝突した樹木表面は成長を促進され、障害物を巻き込むように成長している。成長が進むと完全に障害物を巻き込む様子もシミュレートされている。障害物と衝突した樹木表面はコブ球と巻き込み係数  $B$  によって制御可能である。

図 27 は本手法によって生成された様々な表面形状を持つ樹木である。生成時間とポリゴン数を表 3 に示す。樹木の表面形状をシミュレートすることによって、様々な特徴の樹木表面を持った樹木の形状が生成された。枝ぶりはパラメトリック L-System で、表面の形状はいくつかのパラメータによって制御可能である。細かい枝と葉は、枝の先端周辺にランダムに配

置されている。

表 3 結果画像の生成時間, ポリゴン数, 頂点数

	時間(min)	ポリゴン数	頂点数
(a)	48.4	147937	81718
(b)	7.6	40568	23508
(c)	6.7	56794	32758

## 11. おわりに

本研究では、詳細な表面形状を持った樹木形状を計算によって生成した。樹木形状を生成するために、ポリゴンメッシュの変形を利用した樹木の成長シミュレーションを提案した。樹木表面のコブや樹洞といった様々な要素を成長シミュレーション内で考慮することで、経年変化にともなう複雑な表面形状を持つ樹木形状の生成が可能となった。

本手法の制限として、細かい枝を生成できないことがある。ポリゴンサイズを小さくすれば、生成できる枝のサイズは細かくできる。しかし、ポリゴンサイズを極端に小さくすれば計算量が大きくなり実現不可能な計算時間になる。このような特徴から細かい枝や葉をビルボードで描画しているが、図 26 のように細かい枝や葉が平面上に描画されており、違和感がある。そのため、ビルボードを立体的に組み合わせるなどの手法が必要であると考えられる。

また、本研究の問題点として、パラメータ数が多いことが挙げられる。パラメータを調整するためには、パラメータを変えごとに形状生成の試行が必要であり、試行回数は莫大となる。これを解決するためには、実物の植物の成長から得られる情報を用いて、パラメータ設定を自動化するなどの方策をとる必要がある。

本研究では樹木形状の生成を対象としたが、ポリゴンメッシュの変形を用いた成長シミュレーションは、他の様々な自然物形状の生成にも利用できると考える。

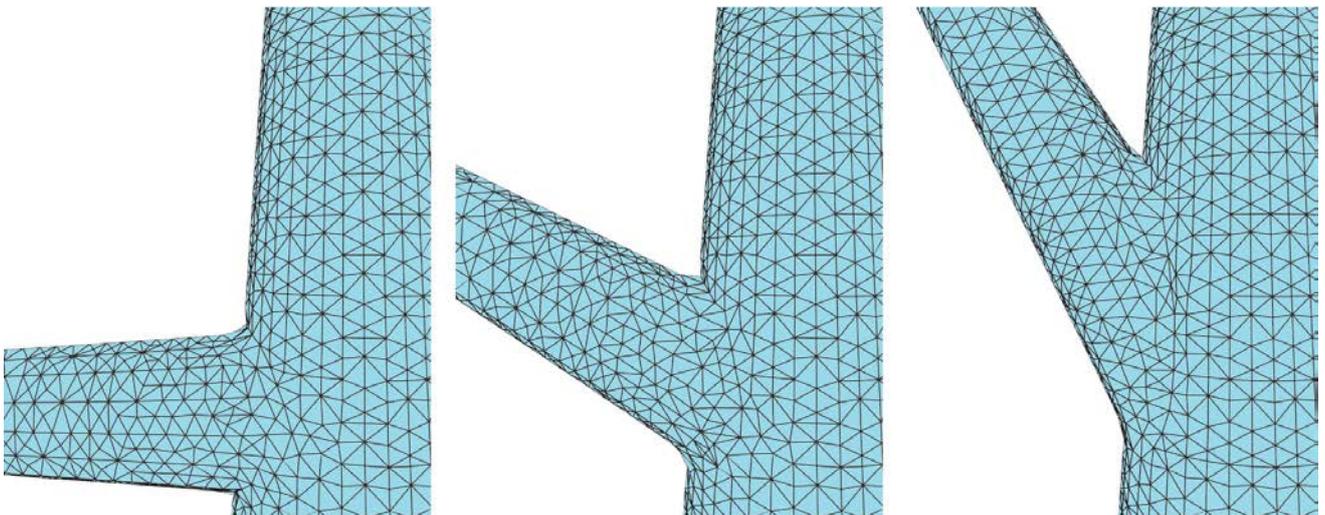


図 22 様々な角度で生成された枝

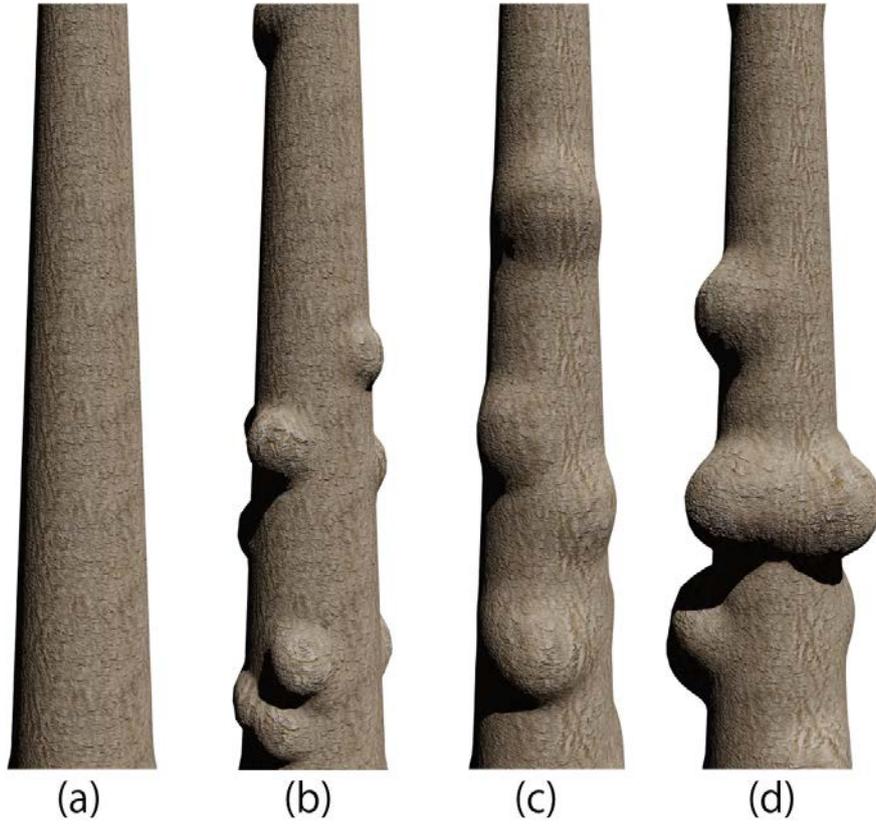


図 23 異常成長によるコブの生成結果

共通パラメータ :  $T_s = 1500\text{mm}$ ,  $T_m = 70\text{mm}$ ,  $S_a = 60\text{mm/step}$ ,  $S_c = 3\text{mm/step}$ ,  $R' = 160\text{mm}$   
(a)コブなし (b) $R=300\text{mm}$ ,  $k=0.5$  (c) $R=600\text{mm}$ ,  $k=0.5$  (d)  $R=600\text{mm}$ ,  $k=1.0$

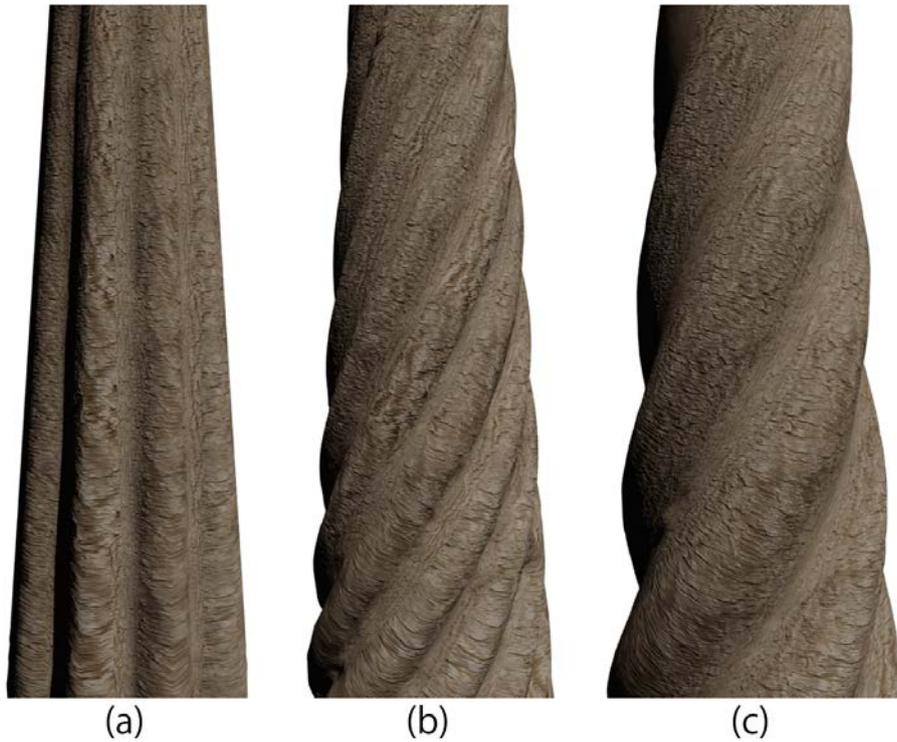


図 24 養分の偏りによるコブの生成結果

共通パラメータ :  $T_s = 150\text{mm}$ ,  $T_m = 70\text{mm}$ ,  $S_a = 60\text{mm/step}$ ,  $S_c = 3\text{mm/step}$ ,  $R' = 160\text{mm}$ ,  $k=1.0$ ,  $n_{end}=300\text{step}$   
(a) $R=3000\text{mm}$ ,  $\theta_p=0\text{rad/mm}$ ,  $m=10$  (b) $R=300\text{mm}$ ,  $\theta_p=0.004\text{rad/mm}$ ,  $m=10$  (c) $R=600\text{mm}$ ,  $\theta_p=0.004\text{rad/mm}$ ,  $m=5$

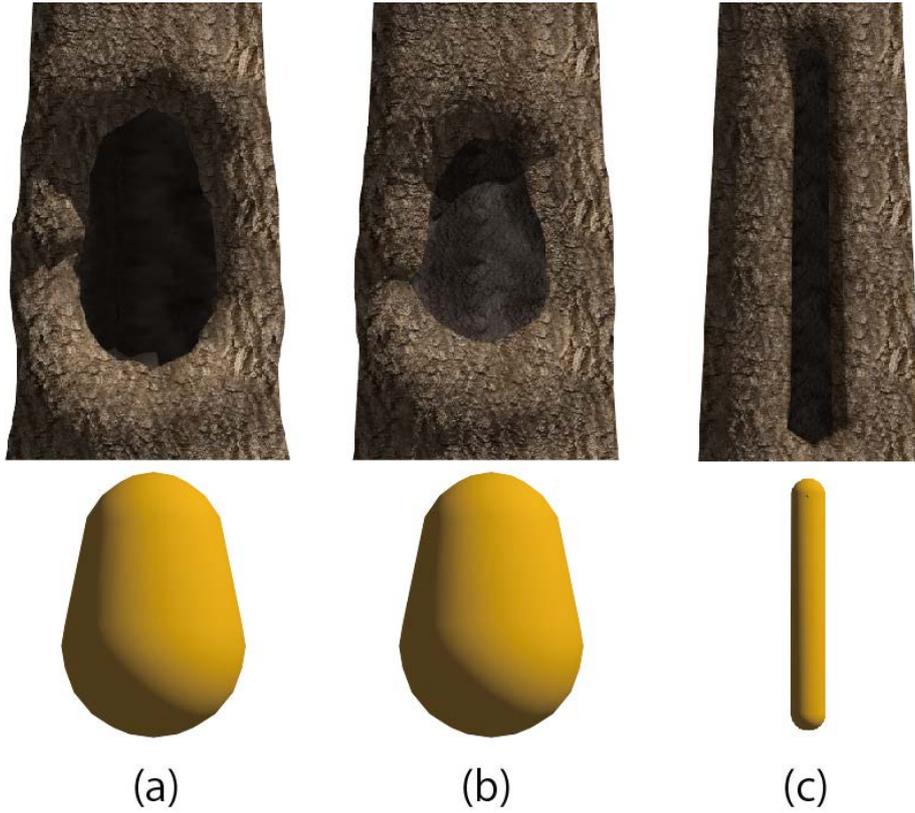


図 25 樹洞の生成結果

共通パラメータ :  $T_s = 150\text{mm}$ ,  $T_m = 70\text{mm}$ ,  $S_a = 60\text{mm/step}$ ,  $S_c = 3\text{mm/step}$ ,  $A = 1$

(a)  $S_h = 6\text{mm/step}$ ,  $R = 250\text{mm}$ ,  $R' = 160\text{mm}$ ,  $k = 0.3$  (b)  $S_h = 0\text{mm/step}$ ,  $R = 250\text{mm}$ ,  $R' = 160\text{mm}$ ,  $k = 0.3$  (c)  $S_h = 6\text{mm/step}$ , コブなし

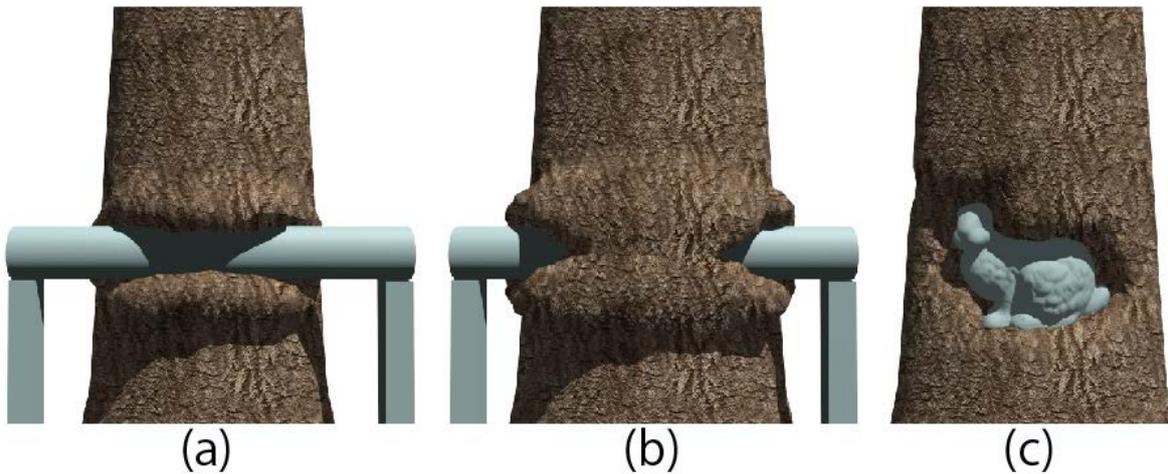


図 26 障害物と衝突した樹木の生成結果

共通パラメータ :  $T_s = 150\text{mm}$ ,  $T_m = 70\text{mm}$ ,  $S_a = 60\text{mm/step}$ ,  $S_c = 3\text{mm/step}$ ,  $B = 1$

(a)  $n_{end} = 350\text{step}$ ,  $R = 300\text{mm}$ ,  $R' = 160\text{mm}$ ,  $k = 30$  (b)  $n_{end} = 400\text{step}$ ,  $R = 300\text{mm}$ ,  $R' = 160\text{mm}$ ,  $k = 30$  (c)  $n_{end} = 400\text{step}$ , コブなし

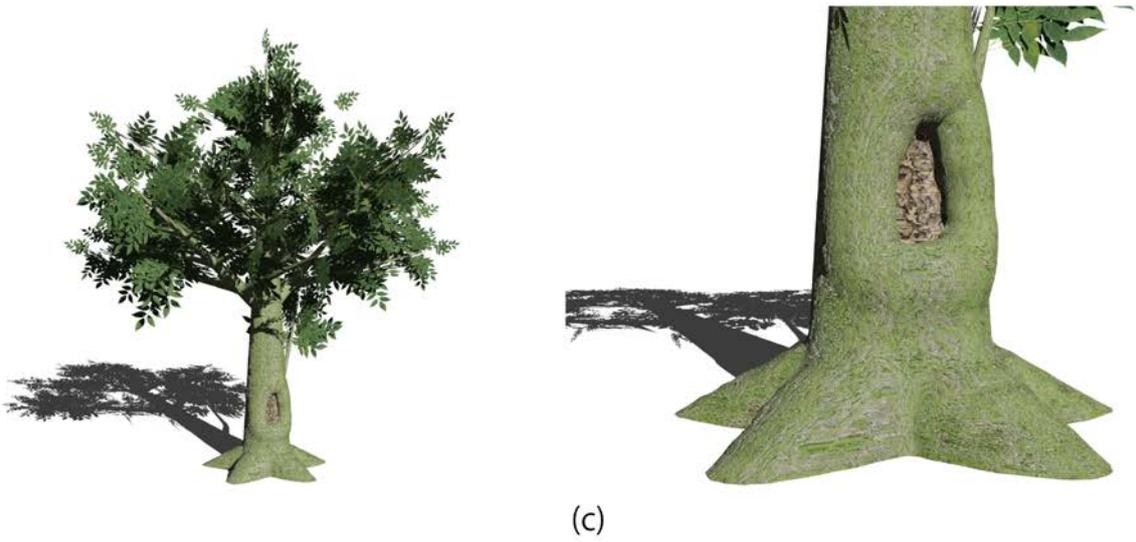
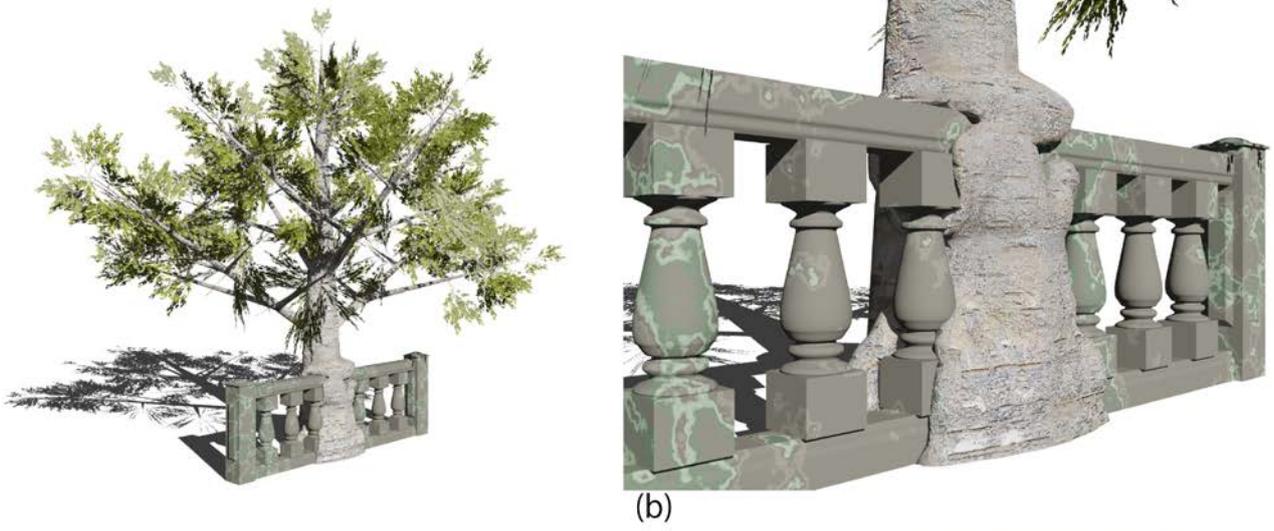
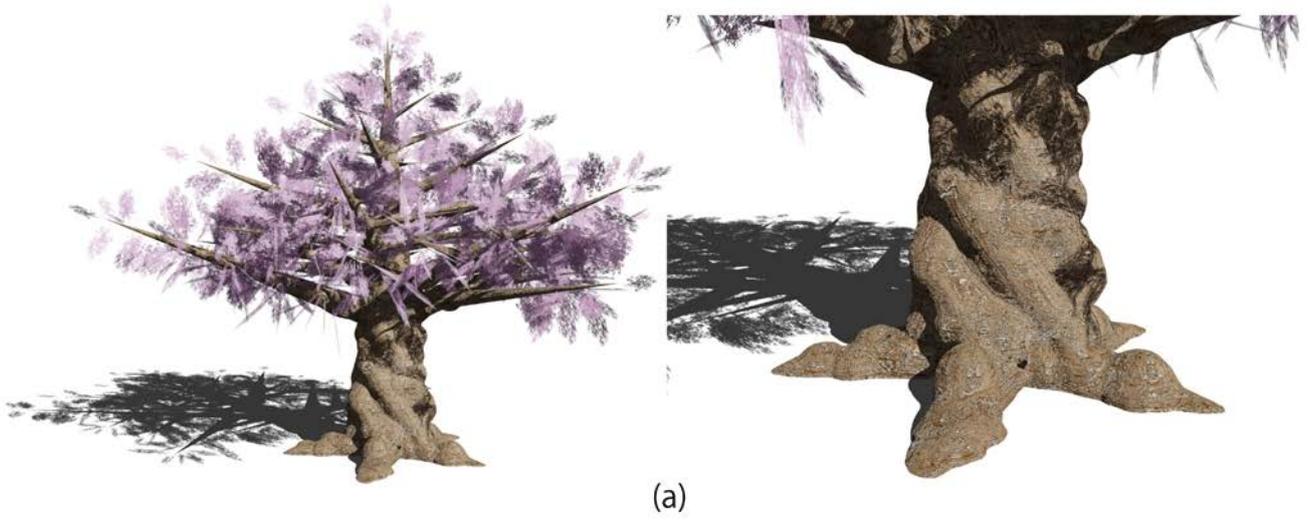


図 27 結果画像

## 参考文献

- [1]S. Ulam: On some mathematical properties connected with patterns of growth of figures. In Proceedings of Symposia on Applied Mathematics, American Mathematical Society. 14, 1962.
- [2]H. Honda: Description of the form of trees by the parameters of the tree-like body: Effects of the branching angle and the branch length on the shape of the tree-like body. *Journal of Theoretical Biology*. 31, 2, 1971.
- [3] A. Lindenmayer: Mathematical models for cellular interaction in development. *Journal of Theoretical Biology* 18, 3, 1968.
- [4] P. Prusinkiewicz, and A. Lindenmayer: *The Algorithmic Beauty of Plants*. Springer-Verlag, 1990.
- [5] P. Prusinkiewicz, M. Hammel, J. Hanan, and R. M ech, L-systems: from the theory to visual models of plants. In Proceedings of the 2nd CSIRO Symposium on Computational Challenges in Life Sciences, 1996.
- [6]P. Prusinkiewicz and J. Hanan: Visualization of botanical structures and processes using parametric L-systems, *Scientific visualization and graphics simulation*, 1990.
- [7] J. C. Hanan: *Parametric L-systems and Their Application to the Modelling and Visualization of Plants*. PhD thesis, The University of Regina, 1992.
- [8] P. Prusinkiewicz, M. James, and R. M ech. Synthetic topiary. In Proceedings of ACM SIGGRAPH 94, 1994.
- [9]R. M ech, and P. Prusinkiewicz: Visual models of plants interacting with their environment. In Proceedings of ACM SIGGRAPH '96, 1996.
- [10]W. Palubicki, K. Horei, S. Longay, A. Runions, B. Lane, R. M ech, and P. Prusinkiewicz: Self-organizing tree models for image synthesis. *ACM Trans. Graph.* 28, 3, 2009.
- [11]S. Pirk, O. Stava, J. Kratt, M. A. M. Said, B. Neubert, R. M ech, B. Benes, and O. Deussen: Plastic trees: interactive self-adapting botanical tree models. *ACM Trans. Graph.* 31, 4, 2012.
- [12]S. Pirk, T. Niese, O. Deussen, and B. Neubert: Capturing and animating the morphogenesis of polygonal tree models. *ACM Trans. Graph.* 31, 6, 2012.
- [13]Z. Cheng, X. Zhang, and B. Chen: Simple reconstruction of tree branches from a single range image. *Journal of Computer Science and Technology* 22, 6, 2007.
- [14]P. Tan, T. Fang, J. Xiao, P. Zhao, and L. Quan: Single image tree modeling. *ACM Trans. Graph.* 27, 5, 2008.
- [15]I. Shlyakhter, M. Rozenoer, J. Dorsey, and S. Teller: Reconstructing 3d tree models from instrumented photographs. *IEEE Comput. Graph.* 21, 3, 2001.
- [16]B. Neubert, T. Franken, and O. Deussen: Approximate image-based tree-modeling using particle flows. *ACM Trans. Graph.* 26, 3, 2007.
- [17]P. Tan, G. Zeng, J. Wang, S. B. Kang, and L. Quan: Image-based tree modeling. *ACM Trans. Graph.* 26, 3, 2007.
- [18]Y. Livny, F. Yan, M. Olson, B. Chen, H. Zhang, and J. El-Sana: Automatic reconstruction of tree skeletal structures from point clouds. *ACM Trans. Graph.* 29, 6, 2010.
- [19]Y. Livny, S. Pirk, Z. Cheng, F. Yan, O. Deussen, D. Cohen-Or, and B. Chen: Texture-lobes for tree modeling. *ACM Trans. Graph.* 30, 4, 2011.
- [20]H. Xu, N. Gossett, and B. Chen: Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Trans. Graph.* 26, 4, 2007.
- [21] M. Okabe, S. Owada, and T. Igarashi: Interactive design of botanical trees using freehand sketches and example-based editing. *Computer Graphics Forum*. 24, 3, 2005.
- [22] T. Ijiri, S. Owada, and T. Igarashi: Sketch L-System: Global Control of Tree Modeling Using Free-form Stroke In Proc of the 6th International Symposium of SmartGraphics, 2006.
- [23] S. Longay, A. Runions, F. Boudon, and P. Prusinkiewicz: TreeSketch: Interactive Procedural Modeling of Trees on a Tablet. Proceedings of the International Symposium on Sketch-Based Interfaces and Modeling, 2012.
- [24]J. Bloomenthal: Modeling the mighty maple. In Proceedings of ACM SIGGRAPH'85, ACM, 1985.
- [25] J. Combaz, and F. Neyret: Painting folds using expansion textures. In Proceedings of Pacific Graphics, 2002.
- [26] J. Combaz, and F. Neyret: Semi-interactive morphogenesis. In Shape Modeling International, 2006.
- [27] T. Ijiri, M. Yokoo, S. Kawabata, and T. Igarashi: Surface-based Growth Simulation for Opening Flowers In Proc. Graphics Interface 2008.
- [28] S. Jeong, S. Park, and C. Kim: Simulation of Morphology Changes in Drying Leaves. *Computer Graphics Forum*. 32, 1, 2013.
- [29]P. Thomas: *Trees: Their Natural History*. Cambridge University Press, 2000.
- [30]R. M. H. Merks, A. G. Hoekstra, J. A. Kaandorp, and P. M. A. Sloot: Polyp oriented modelling of coral growth. *Journal of Theoretical Biology*. 228, 4, 2004.
- [31]K. Schulgasser, and A. Witzuma: The mechanism of spiral grain formation in trees. *Wood Science and Technology*. 41, 2007.

**溝口 敦士**



2006年大阪産業大学工学部機械工学科卒業。2008年北陸先端科学技術大学院大学知識科学研究科知識システム基礎学専攻博士前期課程修了。2010年北陸先端科学技術大学院大学知識科学研究科博士後期課程入学。修士（知識科学）。コンピュータグラフィックスに関する研究に従事。

**宮田 一乗**



1986年東京工業大学大学院・総合理工学研究科・物理情報工学専攻修士課程修了。同年、日本アイビーエム（株）東京基礎研究所入社。1998年東京工芸大学芸術学部助教授。2002年北陸先端科学技術大学院大学知識科学教育研究センター教授。2012年より、同大学知識科学研究科教授。博士（工学）。コンピュータグラフィックスおよびデジタル映像表現に関する研究に従事。情報処理学会、芸術科学会、映像情報メディア学会、ACM、IEEE 等会員。