

4N-2 4元数の3次元空間フラクタルを用いた樹状パターンの生成

松坂 泰洋† 島崎 眞昭‡

†(九州大学工学部) ‡(九州大学大型計算機センター)

1. はじめに

自然物体の一つである樹木のコンピュータグラフィクス表現において、植物のモデリングの方法の研究が盛んになっている^[1]。一方、コンピュータグラフィクスの分野でフラクタルを利用するという考えは自然界のモデルを手軽にかつリアルに表現できるため、最近世界的に注目されてきている^[2]。フラクタルによる生成法はシンプルなアルゴリズムによって形状モデルが生成できる反面、同じ樹種に対して微妙に形状の異なる多くのモデルを得ることが困難である。また、一般に複素力学系においては2次元空間上で取り扱われるためにモデルの生成に要する自己相似縮小写像関数は2次元空間上に限られていた。

一方、最近コンピュータビジョンやロボティクスの分野において空間の位置や運動を表現するための数学的な手法の一つとして、4元数^[3]が注目されてきている。一つの実部と三つの虚部により構成される4元数は、3次元空間において図形の回転、投影、アフィン変換の表現するのに便利であり、生成される図形をイメージするのに直観的にわかりやすい利点がある。

そこで、4元数により表される3次元空間上の自己相似縮小写像関数を用いて3次元空間内に生成される樹木モデルを提案する。このモデルはフラクタルの単純性を有し、かつパラメータによる形状の変化が可能である。

2. 4元数

2.1 4元数

4元数 q は一つの実部 q_r と三つの虚部 q_i, q_j, q_k により構成され、その一般形は

$$q = q_r + q_i i + q_j j + q_k k \quad (1)$$

で表される。 i, j, k は複素単位であり、次の式に従う。

$$i^2 = j^2 = k^2 = ijk = -1. \quad (2)$$

また、 q の絶対値および q^2 は以下のように表される。

$$\|q\| = \sqrt{q_r^2 + q_i^2 + q_j^2 + q_k^2} \quad (3)$$

$$q^2 = q_r^2 - q_i^2 - q_j^2 - q_k^2 + 2q_r(q_i i + q_j j + q_k k). \quad (4)$$

2.2 4元数としての3次元ベクトル

4元数 q について、実部および虚部を

$$q_r = 0, \quad q_i = x, \quad q_j = y, \quad q_k = z, \quad (5)$$

とすると、3次元空間内の点 (x, y, z) は4元数 q により

$$q = ix + jy + kz \quad (6)$$

と表される。ここで、 q が単位ベクトルであるならば、 $q^2 = -1$ である。

4元数を用いる最大の利点は3次元空間における回転の表

Generation of tree pattern using three dimensional space fractals in quaternion
Yasuhiro MATSUSAKA, Masaaki SHIMASAKI
Kyushu University

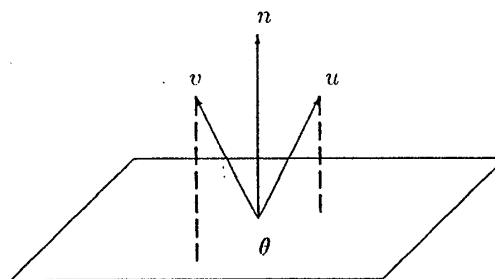


図1: ベクトル n, u, v

現が簡潔であることである。ここで、図1について、回転軸を $n(n_r = 0)$ とする。3次元ベクトル v を、 n のまわり回転角 θ だけ n の上から見て反時計方向に回転して得られるベクトルを u (u, v は4元数。 $u_r = v_r = 0$) とする。このとき、

$$u = qvq^{-1} \quad (7)$$

が成立することが知られている。但し、

$$q = \cos \frac{\theta}{2} + in_i \sin \frac{\theta}{2} + jn_j \sin \frac{\theta}{2} + kn_k \sin \frac{\theta}{2}, \quad (8)$$

$$n_i = \frac{q_i}{\sqrt{q_i^2 + q_j^2 + q_k^2}}, \quad (9)$$

$$n_j = \frac{q_j}{\sqrt{q_i^2 + q_j^2 + q_k^2}}, \quad (10)$$

$$n_k = \frac{q_k}{\sqrt{q_i^2 + q_j^2 + q_k^2}}. \quad (11)$$

また、 i, j, k の性質から次の式が容易に確認される。

$$q^{-1} = \cos \frac{\theta}{2} - in_i \sin \frac{\theta}{2} - jn_j \sin \frac{\theta}{2} - kn_k \sin \frac{\theta}{2}. \quad (12)$$

3. 3次元空間フラクタルによる樹状パターンの生成

図2において、パラメータ $\alpha_m, \beta_m, \gamma_0, \gamma_m (m \geq 1)$ は0から1の間の値をとり、 $r_A, r_{B_0}, r_{B_m}, r_{C_m}, r_D$ はそれぞれ点 A, B_0, B_m, C_m, D の幹の太さを表し、 $\lambda_{AD}, \lambda_{B_m C_m}$ はそれぞれ点 A と D 、点 B_m と C_m の幹の太さの縮小比率を表す。

このとき、自己相似縮小写像関数 f_0 は次の写像変換を行なう。

- z 軸のまわりの回転角 ξ_0 による回転
- 縮小比率 $1 - \gamma_0$ による縮小
- z 軸上の平行移動

これから、 $f_0(v)$ は

$$f_0(v) = (1 - \gamma_0)q_{\xi_0} v q_{\xi_0}^{-1} + k\gamma_0. \quad (13)$$

次に、 $m(m=1,2,\dots)$ 番目の枝に関する自己相似縮小写像

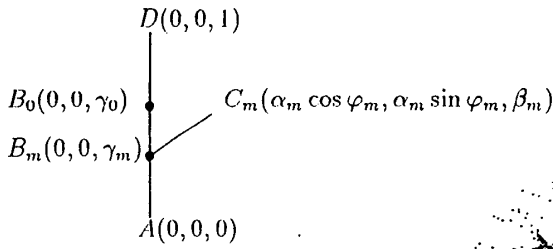


図 2: 樹状パターンの初期図形

関数 f_m は次の写像変換を行なう。

- z 軸のまわりの回転角 ξ_m による回転
- y 軸のまわりの回転角 θ_m による回転
- z 軸のまわりの回転角 φ_m による回転
- 縮小比率 $\sqrt{\alpha_m^2 + (\beta_m - \gamma_m)^2}$ による縮小
- z 軸上の平行移動

これから、 $f_m(v)$ は

$$f_m(v) = \sqrt{\alpha_m^2 + (\beta_m - \gamma_m)^2} q_{\varphi_m} q_{\theta_m} q_{\xi_m} v q_{\xi_m}^{-1} q_{\theta_m}^{-1} q_{\varphi_m}^{-1} + k\gamma_m. \quad (14)$$

但し、

$$\begin{aligned} v &= ix + jy + kz, & (15) \\ q_{\xi_m} &= \cos \frac{\xi_m}{2} + k \sin \frac{\xi_m}{2}, & (16) \\ q_{\theta_m} &= \cos \frac{\theta_m}{2} + j \sin \frac{\theta_m}{2}, & (17) \\ q_{\varphi_m} &= \cos \frac{\varphi_m}{2} + k \sin \frac{\varphi_m}{2}, & (18) \\ \theta_m &= \tan^{-1} \frac{\beta_m - \gamma_m}{\alpha_m}. & (19) \end{aligned}$$

なお、各点の幹の太さは、 f_0 もしくは f_m による写像変換が行われる度にそれぞれ次のように縮小される。

$$r'_A = r_{B_0}, \quad (20)$$

$$r'_{B_0} = \lambda_{AD} r_{B_0}, \quad (21)$$

$$r'_{B_m} = \lambda_{B_m C_m} r_{B_m}, \quad (22)$$

$$r'_{C_m} = \lambda_{B_m C_m} r_{C_m}, \quad (23)$$

$$r'_D = \lambda_{AD} r_D \quad (24)$$

以上に挙げた f_0, \dots, f_m を使い、初期図形 K_0 に対して

$$K_{i+1} \leftarrow f_0(K_i) \cup \dots \cup f_m(K_i) \quad (i = 0, 1, 2, \dots) \quad (25)$$

のように反復適用することにより多様な樹状パターンを描くことが可能になる。

4. 実験評価

上に示した自己相似縮小写像関数を利用して、実験を行った。図 3 はその結果の一つである。関数は f_1, f_2 を使用し、反復回数は 8 回にして、パラメータは以下のように選んだ。

- f_1 : $\alpha_1 = 0.3, \beta_1 = 0.9, \gamma_1 = 0.25, \xi_1 = 80^\circ, \varphi_1 = 0^\circ$
- f_2 : $\alpha_2 = 0.6, \beta_2 = 0.7, \gamma_2 = 0.25, \xi_2 = 80^\circ, \varphi_2 = 180^\circ$
- 共通: $r_A = 0.03, r_{B_1} = 0.02, r_{B_2} = 0.02, r_{C_1} = 0.015,$
 $r_{C_2} = 0.015, \lambda_{AD} = 0.75, \lambda_{B_1 C_1} = 0.6, \lambda_{B_2 C_2} = 0.6$

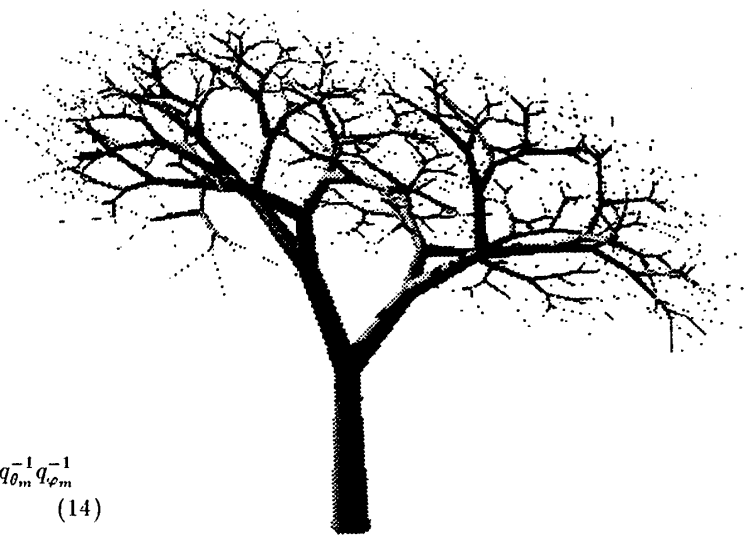


図 3: 実験結果例

パラメータの選定については、実物の樹木の写真をもとにし、試行錯誤して決定した。その結果、実物と比較しても遜色のない結果が得られた。この図から、4 元数による 3 次元フラクタルを用いることでリアルな結果が得られることが分かる。また、パラメータを変化させることにより、形状の異なるモデルを描くことができる。

5. おわりに

今回 4 元数による 3 次元空間フラクタルを用いて、生成される樹状パターンを示した。2 次元空間フラクタルを回転して得られる回転対称の 3 次元モデルと違って、この樹状パターンは 3 次元的複雑さを持っており、視点の移動に伴う景観の変化も表現される。また、パラメータを設定し、それを変えることによって樹木の形状変化が可能になった。今後の課題としては、樹木に葉をつけた場合についての実験を行なうことが考えられる。

参考文献

- [1] 安居院 猛, 福田智美, 中嶋正之: 景観表示のための樹木の生成方法. 情報処理学会論文誌, Vol.32, No.5, pp. 618-625, May, 1991.
- [2] Peter E. Oppenheimer: Real Time Design and Animation of Fractal Plants and Trees. Computer Graphics, Vol.20, No.4, pp. 55-64, 1986.
- [3] Edward Pervin, Jon A. Webb: QUATERNIONS IN COMPUTER VISION AND ROBOTICS. Carnegie Mellon University, Computer Science technical report, 1982, 150.