Tạp chí Tin học và Điều khiển học, T. 15, S. 3 (1999), 66-72

XÂY DỰNG CHƯƠNG TRÌNH THIẾT KẾ ẢNH FRACTAL TRỰC QUAN

NGUYỄN XUÂN HUY, NGUYỄN XUÂN HOÀI

Abstract. Fractal geometry is considered to be one of new fields in mathematics with many applications. This paper proposes a method to build a program (called VFDESIGN) in order to help the process of designing fractal images visually on computer screens. Also, we explain how this program can help users to find out fractal codes for natural objects (to solve the inverse problem).

1. GIỚI THIỆU

Hình học fractal bắt đầu được biết đến từ năm 1975 qua một bài báo mà sau này trở thành một cuốn sách của Benoit Mandelbrot [6]. Từ đó đến nay, chỉ trong một thời gian ngắn đã có nhiều ứng dụng của hình học fractal được tìm ra (xin xem trong [3], [8]). Một trong những ứng dụng được nhiều người biết đến là khả năng mô phỏng các đối tượng thiên nhiên với độ phức tạp cao như cây cối, hoa lá,... (bài toán xuôi) và khả năng nén ảnh (bài toán ngược). Nếu như việc vẽ hình trong hình học Euclide tương đối đơn giản, chúng ta chỉ cần dùng compa, bút chì và thước kẻ, thì đối với hình học fractal chỉ có máy tính mới có thể giúp cho chúng ta thấy được hình ảnh hình học (xấp xỉ) của một tập fractal do khối lượng tính toán lớn. Trong bài báo này chúng tôi xin đưa ra một phương pháp xây dựng một chương trình hỗ trợ thiết kế fractal trực quan phục vụ cho công tác mô phỏng hình học trên máy tính và giúp đỡ cho người dùng tìm ra mã fractal của một đối tượng bất kỳ (giải bài toán ngược). Phần 2 dưới đây sẽ trình bày về lý thuyết hệ hàm lặp làm cơ sở cho việc vẽ hình fractal. Phần 3 đề cập tới việc thiết kế chương trình VFDESIGN, một chương trình hỗ trợ thiết kế ảnh fractal trên máy tính với các phép biến đổi fractal là các phép biến đổi afin. Phần 4 bao gồm một số nhận xét, kết luận và hướng phát triển. Trong phần này cũng có mô tả cách sử dụng chương trình VDFDESIGN để trợ giúp cho việc giải bài toán ngược.

2. CƠ SỞ LÝ THUYẾT VỀ HỆ HÀM LẶP

Có nhiều mô hình cho việc tạo ảnh fractal, tuy nhiên nội dung bài báo chỉ giới hạn trong mô hình hệ hàm lặp (IFS - Iterated Function System) và mô hình hệ hàm lặp có gắn xác suất (FIFS - Fortune Wheel Iterated Function System) [2]. Phần này sẽ lần lượt trình bày về nguyên lý điểm bất động của ánh xạ co - cơ sở toán học cho các hệ hàm lặp, mô hình IFS FIFS và các thuật toán tạo sinh (vẽ hình) fractal tương ứng.

2.1. Ánh xạ co ([10])

Một ánh xạ $f : (X, d) \to (X, d)$ *trong đó* (X, d) *là một không gian metric được gọi là co nếu* $\exists s,\ 0 \le s < 1$ *sao cho* $\forall x, y \in X$:

$$d(f(x), f(y)) \le s \times d(x, y), \tag{1}$$

s được gọi là độ co của f.

2.2. Nguyên lý điểm bất động của ánh xạ co ([10])

Giả sử (X, d) *là một không gian metric đầy đủ,* f *là một ánh xạ co trên* (X, d)*, khi đó tồn tại duy nhất một điểm bất động* x_f *sao cho:*

$$x_f = f(x_f) = \lim_{n\to\infty} f^{(n)}(x_0) \ \forall x_0 \in X, \tag{2}$$

trong đó $f^{(n)}(x_0) = x_n$ *với* $x_i = f(x_{i-1}),\ i = 1, 2, \ldots$

Chứng minh. Xem trong [10] hoặc [4]. □

2.3. Định lý Collage ([1])

Giả sử (X, d) là một không gian metric đầy đủ, f là một ánh xạ co với độ co s và x_f là điểm cố định của f thì ta có:

$$d(x, x_f) \le \frac{1}{1-s} \times d(x, f(x)) \ \forall x \in X. \tag{3}$$

Chứng minh. Xem trong [1] hoặc [4]. □

2.4. Mô hình hệ hàm lặp (IFS)

2.4.1. Định nghĩa không gian $(\mathcal{H}(X), h)$ ([1])

Cho không gian metric (X, d) ta ký hiệu $\mathcal{H}(X)$ là tập tất cả các tập compact khác rỗng của X. Ánh xạ $h : \mathcal{H}(X) \times \mathcal{H}(X) \to R$ được định nghĩa như sau:

$$h(A, B) = \max\left(d(A, B), d(B, A)\right), \text{ trong đó } d(A, B) = \max_{a\in A}\left(\min_{b\in B} d(a, b)\right). \tag{4}$$

Trong [4] Michael Barnsley đã chứng minh rằng h là một metric trên $\mathcal{H}(X)$ và hơn thế nữa nếu (X, d) là một không gian metric đầy đủ thì $(\mathcal{H}(X), h)$ là một không gian metric đầy đủ.

2.4.2. Định nghĩa IFS ([1])

Cho một không gian metric đầy đủ (X, d), ta định nghĩa một $IFS = \{w_1, w_2, \ldots, w_n\}$ trong đó các w_i là các ánh xạ co trên (X, d) với độ co s_i, $i = 1, 2, \ldots, n$.

Khi đó toán tử Hutchinson $W : (\mathcal{H}(X), h) \to (\mathcal{H}(X), h)$ được định nghĩa như sau:

$$W(A) = \bigcup_{i=1}^{n} w_i(A). \tag{5}$$

Định lý 2.4.1. *Toán tử Hutchinson là một ánh xạ co trên không gian $(\mathcal{H}(X), h)$ với độ co $s = \max_{1\le i\le n} s_i$.*

Chứng minh. Xem trong [1] hoặc [8]. □

Như vậy theo phần 2.2, W có một điểm bất động là $\mathcal{A} \in \mathcal{H}(X)$. Lúc đó ta gọi $\mathcal{A}$ là điểm bất động của IFS, hay $\mathcal{A}$ là tập fractal tạo sinh bởi IFS và ngược lại IFS là mã fractal của $\mathcal{A}$.

2.4.3. Bài toán sinh

Cho một IFS, hãy tìm điểm bất động (tập fractal) $\mathcal{A}$.

Theo phần 2.2 ta thấy để giải bài toán sinh ta chỉ việc tìm $\mathcal{A} = A_\infty = \lim_{n\to\infty} A_n$. Trong đó $A_i \in \mathcal{H}(X)$, $i = 0, 1, 2, \ldots$, $A_i = W(A_{i-1})$, $i = 1, 2, \ldots$, A_0 chọn bất kỳ, với W là toán tử Hutchinson đại diện cho IFS. Tuy nhiên trên máy tính chúng ta không thể tìm A_∞ thay vào đó ta tìm A_M với M nào đó (trong thực tế lập trình thường chọn $M \in [8..15]$). Việc vẽ ảnh fractal dựa trên IFS (theo bài toán sinh) được thực hiện bởi thuật toán đệ quy sau đây:

+ Thuật toán vẽ hình fractal với IFS: (với $X = R^2$ hoặc $X = R^3$).

Vào: IFS $= \{w_1, w_2, \ldots, w_n\}$, một đối tượng xuất phát Ob bất kỳ thuộc $\mathcal{H}(X)$, D (độ sâu đệ quy, quá trình đệ quy dừng khi $D = M$).

Ra: Hình vẽ A_M xấp xỉ $\mathcal{A}$.

```
Procedure Generate (Ob: Object; D: Interger);
   var i: integer;
      Begin
         if D = M then
            Draw (Ob)
         else
            for i := 1 to n do
               Generate (w_i (Ob), D + 1);
      End;
```

Để vẽ hình fractal tạo sinh bởi IFS ta chỉ cần gọi Generate (Ob, 0). (Thủ tục có độ phức tạp thời gian mũ theo số phép biến đổi).

2.4.4. Bài toán ngược

Cho $\mathcal{A} \in \mathcal{H}(X)$, tìm IFS nhận $\mathcal{A}$ làm điểm cố định.

Việc giải bài toán ngược có ứng dụng trong việc nén ảnh vì độ phức tạp thông tin để lưu trữ IFS thường nhỏ hơn rất nhiều lần so với độ phức tạp của ảnh $\mathcal{A}$. Việc giải quyết bài toán ngược có thể được hỗ trợ bởi định lý Collage cho IFS:

Định lý 2.4.2 (Định lý Collage cho IFS) ([1]).

$$h(A, \mathcal{A}) \leq \frac{1}{1-s} \times h(A, W(A)) \;\; \forall A \in \mathcal{H}(X). \tag{6}$$

Như vậy để tìm IFS có điểm cố định $\mathcal{A}$ xấp xỉ A cho trước ta chỉ cần tìm IFS sao cho $h(A, W(A))$ là đủ nhỏ (với điều kiện $1/(1-s)$ là nhỏ). Nếu coi A là một ảnh thì thay vì phải lưu trữ A ta chỉ cần lưu trữ IFS có điểm cố định xấp xỉ A (do đó phương pháp nén ảnh fractal thuộc lớp các phương pháp nén mất mát).

2.5. Mô hình hệ hàm lặp có gắn xác suất (FIFS - [5])

2.5.1. Định nghĩa không gian $(\mathcal{P}(X), d_H)$ ([5])

Giả sử (X, d) là một không gian metric compact, ta định nghĩa: $\mathcal{P}(X) = \{\mu \mid \mu$ là một độ đo chuẩn hóa trên σ-đại số Borel $\mathcal{B}(X)$ của $X\}$. Ánh xạ $d_H : \mathcal{P}(X) \times \mathcal{P}(X) \to R$ *được xác định như sau:*

$$d_H(\mu, \nu) = \sup \left\{ \int_X f d\mu - \int_X f d\nu \mid f : X \to R, \; f \text{ liên tục và } |f(x) - f(y)| \leq d(x, y), \forall x, y \in X \right\} \tag{7}$$

Hutchinson trong [5] đã chứng minh rằng d_H là một metric trên $\mathcal{P}(X)$ hơn thế nữa $(\mathcal{P}(X), d_H)$ là một không gian metric compact.

2.5.2. Định nghĩa FIFS ([5])

Cho không gian metric compact (X, d), ta định nghĩa một FIFS $= \{w_1, w_2, \dots, w_n; p_1, p_2, \dots, p_n\}$ trong đó: với $i = 1, 2, \dots, n$, w_i là ánh xạ co trên không gian metric (X, d), p_i là xác suất gắn với w_i, $0 < p_i < 1$, $\sum_{i=1}^{n} p_i = 1$.

Khi đó ta định nghĩa toán tử Markov M: $(\mathcal{P}(X), d_H) \to (\mathcal{P}(X), d_H)$ như sau:

$$M(\nu) = \sum_{i=1}^{n} p_i \times \nu \circ w_i^{-1}. \tag{8}$$

Định lý 2.5.1 ([5]). *Toán tử Markov M là một ánh xạ co trên $(\mathcal{P}(X), d_H)$ với độ co s của IFS $= \{w_1, w_2, \dots, w_n\}$. Do đó tồn tại một độ đo bất biến μ^* sao cho $M(\mu^*) = \mu^*$.*

Chứng minh. Xem trong [1] hoặc [5].

Định lý 2.5.2 (Định lý Collage cho FIFS) ([5])

$$d_H(\nu, \mu^*) \le \frac{1}{1-s} d_H(\nu, M(\nu)). \tag{9}$$

+ Thuật toán vẽ hình Fractal với FIFS:

Vào: FIFS=$\{w_1, w_2, ..., w_n; p_1, p_2, ..., p_n\}$, điểm xuất phát (x_0, y_0) bất kỳ thuộc R^2, số lần lặp N.
Ra: Hình Fractal A_N xấp xỉ A của IFS.

```
Begin
    (x, y) := (x0, y0);
      For i := 1 to N do
          Begin
            chọn wj với xác suất pj;
            (x, y) := wj(x, y);
              if i > M then {Bỏ qua M điểm đầu không vẽ}
                Vẽ (x, y);
          End;
End;
```

(Thuật toán có độ phức tạp tuyến tính theo số lần lặp. Trong thực tế N thường được chọn khoảng 30.000, M khoảng 200).

3. XÂY DỰNG CHƯƠNG TRÌNH THIẾT KẾ FRACTAL TRỰC QUAN

Trong phần này chỉ xét đến ánh xạ afin trong không gian định chuẩn R^n. Chúng có dạng như sau:

$$f(X) = AX + b. \tag{10}$$

Trong đó X và b là các véctơ trong R^n, A là toán tử tuyến tính trên R^n. Có thể biểu diễn A bằng một ma trận (a_{ij}) $i = 1, \dots, n$; $j = 1, \dots, n$.

Mệnh đề 3.1. *Nếu $\|A\| < 1$ thì ánh xạ afin f là co trên (R^n, d) (trong đó $d(x, y) = \|x - y\|$ $\forall x, y \in R^n$).*

Chứng minh. Hiển nhiên, ánh xạ afin là co khi và chỉ khi toán tử tuyến tính A là một ánh xạ co. Theo kết quả về toán tử tuyến tính ([10]) chúng ta có: $\forall x, y \in R^n d(A(x), A(y)) = \|A(x) - A(y)\| = \|A(x-y)\| \le \|A\| \times \|x - y\| = \|A\| \times d(x, y)$. Vì vậy nếu $\|A\| < 1$ thì ta có ánh xạ afin là co. □

Có nhiều cách xác định $\|A\|$, ví dụ: $\|A\| = \max\limits_{1 \le j \le n} \big(\sum\limits_{i=1}^{n} |a_{ij}|\big)$.

Trong trường hợp $n = 2$ (10) có dạng như sau:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}. \tag{11}$$

Mệnh đề 3.2. *Giả sử $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ là ma trận của phép biến đổi tuyến tính không suy biến trên R^2, khi đó ta có:*

$$A = A_s A_t A_u A_\theta. \tag{12}$$

Trong đó:

$$A_s = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \textit{ (ma trận của phép giãn) với } s = \frac{ad - bc}{\sqrt{c^2 + d^2}},$$

$$A_t = \begin{bmatrix} 1 & 0 \\ 0 & t \end{bmatrix} \textit{ (ma trận của phép giãn theo trục y) với } t = \frac{c^2 + d^2}{ad - bc},$$

$$A_u = \begin{bmatrix} 1 & u \\ 0 & 1 \end{bmatrix} \textit{ (ma trận của phép kéo nghiêng theo trục x) với } u = \frac{ac + bd}{ad - bc},$$

$$A_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \textit{ (ma trận của phép quay) với } \theta = \arccos\left(\frac{d}{\sqrt{c^2 + d^2}}\right).$$

Chứng minh. Dùng phép nhân ma trận thông thường. □

Như vậy một phép biến đổi afin trên mặt phẳng có thể coi là tương đương với 5 phép biến đổi: giãn, giãn theo trục y, kéo nghiêng theo trục x, quay và dịch chuyển (hệ số e, f).

Vì vậy một chương trình hỗ trợ thiết kế fractal trực quan trên R^2 với các phép biến đổi afin sẽ giúp người dùng thiết kế mã (hình ảnh) fractal như sau:

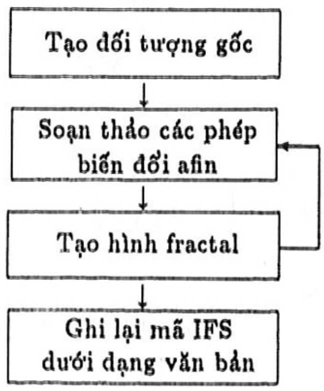
- Người thiết kế chọn một đối tượng xuất phát, gọi là đối tượng gốc. Mặc dù đối tượng gốc có thể chọn bất kỳ nhưng trong chương trình VFDESIGN chúng tôi lựa chọn là hình vuông đơn vị: $[0, 1] \times [0, 1]$.

- Người thiết kế nhập các phép biến đổi afin để biến đổi đối tượng gốc (tạo ra các đối tượng con). Chương trình cho phép người thiết kế nhập dữ liệu trực tiếp từ bàn phím (các hệ số a, b, c, d, e, f), hoặc chọn một phép biến đổi đã có sẵn trong năm phép biến đổi trên. Khi một phép biến đổi đang được thực hiện thì kết quả luôn hiện ra (tính trực quan).

- Sau khi người thiết kế nhập xong các phép biến đổi (đối tượng) chương trình cho người thiết kế xem kết quả (hình fractal) bằng việc kiểm tra tính co của IFS (theo Mệnh đề 3.1) sau đó thực hiện một trong hai thuật toán được nêu trong mục 2 (với độ sâu đệ quy từ 8 đến 15 cho thuật toán IFS. Số lượng điểm 20.000 - 40.000, điểm xuất phát (0, 0), bỏ qua khoảng 200 điểm đầu đối với thuật toán FIFS). Các giá trị xác suất p_i $(i = 1, 2, \ldots, n)$ cho FIFS được lựa chọn như sau: $p_i = \max\left(\frac{|\det A_i|}{\sum_{j=1}^{n} |\det A_j|}, 0{,}001\right)$ (các phép biến đổi suy biến được gán xác suất = 0,001 và phải nhập các hệ số a, b, c, d, e, f từ bàn phím (giá trị 0,001 được chọn dựa trên thực nghiệm).

- Nếu người thiết kế nhận thấy hình tạo ra chưa đúng với yêu cầu mô phỏng thì họ có thể dùng phương trình để sửa lại các phép biến đổi (thêm phép biến đổi, xóa phép biến đổi, thay đổi phép biến đổi) và tiếp tục tạo sinh cho đến khi đạt yêu cầu.

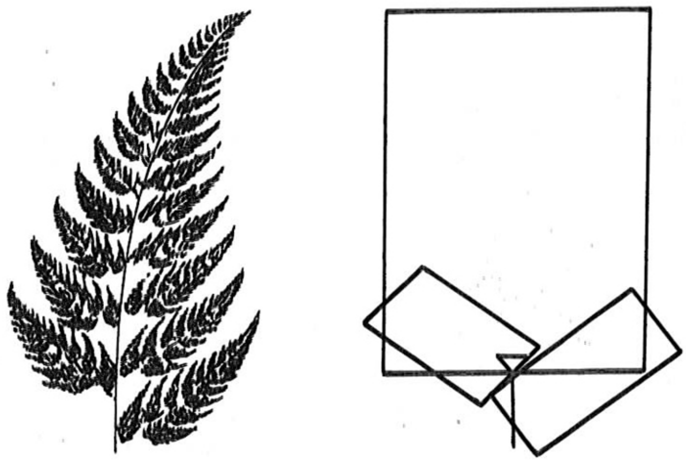
- Khi thiết kế xong hình fractal chương trình cho phép người thiết kế ghi lại hình fractal (dưới dạng văn bản - các hệ số của IFS - do đó dung lượng lưu trữ cho ảnh sẽ rất nhỏ so với việc lưu giữ các điểm ảnh).



Hình 1. Quá trình thiết kế fractal

4. KẾT LUẬN, NHẬN XÉT, HƯỚNG PHÁT TRIỂN

Chương trình VFDESIGN phiên bản 1.0 được xây dựng trên hệ điều hành MSDOS có khả năng cho phép soạn, sửa và xem các hình fractal một cách trực quan. Chương trình cho phép người dùng lưu trữ lại ảnh fractal tạo sinh dưới dạng văn bản và dạng ảnh bit map. Ví dụ sau đây là lá dương xỉ của Michael Barnsley bên phải là mã fractal và bên trái là ảnh tương ứng được phát sinh bởi chương trình với thuật toán FIFS, số lần lặp là 30000.



Để vẽ mô hình mô phỏng đối tượng trong hình học Euclide chúng ta phải quan sát đối tượng đó (nhớ toàn bộ) rồi dùng thước kẻ, compa và bút chì sao chép lại nguyên bản theo từng đường nét, chính vì vậy mà hình học Euclide gặp khó khăn trong việc mô phỏng các đối tượng phức tạp đặc biệt là trong thiên nhiên như cây, lá đường bờ biển... Việc vẽ hình một đối tượng fractal ngoài một chương trình hỗ trợ thiết kế fractal như VFDESIGN người thiết kế còn cần biết cách nhìn các đối tượng dưới con mắt của Định lý Collage. Ví dụ như khi nhìn lá dương xỉ nói trên người thiết kế phải biết thực chất lá dương xỉ này là sự chắp nối của lá trái, lá phải, lá trên, cán cây là phiên bản biến đổi từ chính nó (tính tự tương tự). Từ đó xác định các phép biến đổi và dùng VFDESIGN để chỉnh sửa và kiểm tra và do đó có thể tìm ra mã fractal tạo sinh lá dương xỉ (giải bài toán ngược). Chúng tôi xin đề đạt công thức để tìm mã tạo sinh fractal như sau:

Định lý Collage + Sự trợ giúp của VFDESIGN = Tìm ra mã tạo sinh fractal.

Các hướng phát triển cho chương trình VFDESIGN sắp tới là: mô phỏng fractal bằng các mô hình hệ hàm lặp khác như IFZS (hệ hàm lặp để tìm fractal trên các tập mờ), IFSM (hệ hàm lặp với các ánh xạ mức xám đi kèm). Tự tìm mã fractal cho người dùng (nén ảnh đa cấp xám), thêm các công cụ tính chiều, phân đoạn ảnh fractal.

TÀI LIỆU THAM KHẢO

[1] M. F. Barnsley, *Fractal Every Where*, Academic Press, 1988.

[2] M. F. Barnsley, S. Demko, Iterated function system and the global construction of fractals, *Processings of the Royal Society of London* **A399** (1985).

[3] A. Bunde, S. Havlin (eds), *Fractal in Science*, Springer-Verlag, 1994.

[4] Y. Fisher (ed), *Fractal Image Comperession: Theory and Application*, Springer-Verlag, 1995.

[5] J. Hutchinson, Fractals and sefl-similarity, *Indiana University Journal of Mathematics* **2** (1985).

[6] B. Mandelbrot, *The Fractal Geometry of Nature*, W. H. Freeman and Co., 1982.

[7] Nguyễn Quốc Toản, *Giáo trình đại số tuyến tính*, Đại học Tổng hợp Hà Nội, 1980.
[8] Nguyễn Xuân Hoài, *Hình học Fractal và Ứng dụng*, Luận văn thạc sĩ, chuyên ngành: Đảm bảo toán học cho máy tính và các hệ thống tính toán, Đại học Khoa học tự nhiên, Đại học Quốc gia Hà Nội, 1997.
[9] H. O. Peigen, H. Jugens, D. Saupe, *Chaos and Fractal: New Frontiers of Science*, Springer-Verlag, 1992.
[10] Phan Đức Chính, *Giải tích hàm*, Đại học và Trung học chuyên nghiệp, 1978.
[11] W. Rudin, *Functional Analysis*, Mc GrawHill, 1973.

Nhận bài ngày 31 - 3 - 1998
Nhận lại sau khi sửa ngày 22 - 7 - 1999

Nguyễn Xuân Huy – Viện Công nghệ thông tin.
Nguyễn Xuân Hoài - Khoa Toán - Tin học,
Học viện Kỹ thuật Quân sự.