

一类几何分形的表示模型

廖士中

(辽宁师范大学计算机科学系, 大连 116029)

石纯一

(清华大学计算机科学与技术系, 北京 100084)

摘要 结合迭代函数系统和有限自动机,提出了几何分形的正则生成系统表示模型,并应用生成测试法和自动机归纳学习算法研究了一类几何分形的建模方法。实验结果表明,该模型是简单有效的。

关键词 几何分形 分形建模 自动机归纳学习 形状表示

0 引言

分形几何为表示自然物体和景物提供了有效的手段,并在纹理分割、图象压缩、自然景物绘制和识别等领域得到了应用,自提出后,分形表示模型的研究一直受到重视。

80年代由 Hutchinson、Barnsley 等发展起来的迭代函数系统(IFS),是解析地表示分形的数学模型,为分形的表示、分析、绘制等奠定了理论基础^[1]。1988年 Prusinkiewicz 等提出的 Koch 系统,利用向量模式的推导来逼近、表示分形曲线^[2]。1993年 Gujar 等在研究几何分形的遍历时使用了类似的方法^[3]。1992年 Prusinkiewicz 等将形式语言方法同迭代函数系统结合起来,提出了语言受限的迭代函数系统(LRIFS),用于绘制具有部分对称结构的分形^[4]。另外 L-系统也可用于表示分形^[5]。

分形可分为3类:代数分形、随机分形和几何分形^[3]。在几何分形的表示方面,Koch 系统不能表示部分对称性;LRIFS 虽可绘制出具有部分对称结构的分形,但其形状基元是点,形状表示能力较弱;L-系统没有明确表示出部分对称结构,因而难于应用。这些工作都没有考虑建模问题。

本文以迭代函数系统的有关理论为基础,在 Prusinkiewicz 和 Gujar 等工作的基础上,给出了一类

几何分形的表示形式,并应用 Porat 的顺序示例集自动机归纳学习算法,研究了相应的建模技术。本文所给出的几何分形表示模型简单有效,并有相应的建模方法,是一项完整的工作。

1 正则生成系统

几何分形可由几何模式推导出来。初始的几何模式称为初始子,推导可由产生式规则描述。初始子或推导出的几何模式中的所有或部分形状基元不断置换为产生式后件模式,就可生成一个几何分形。显然,几何分形具有有限的粒度范围和部分对称的结构,见图1。

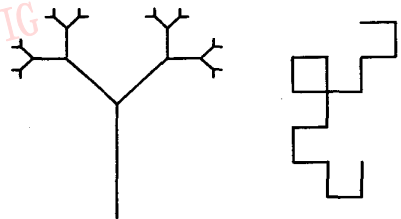


图1 几何分形例子

根据几何分形的特点,本文提出正则生成系统表示模型,其基本思想是:

- (1) 以向量作为形状表示基元;
- (2) 以产生式规则表示向量变换规则,并以推

导长度描述表示的详略程度；

(3) 产生式规则建立不同层次表示之间的关系；

(4) 以有限自动机描述部分对称结构。

下面首先约定一些记号,然后介绍几何分形的正则生成系统表示模型。

(1) 向量的集合 $V = \{v_1, v_2, \dots, v_n\}$ 为 $V = v_1 v_2 \dots v_n$,

(2) V 中各向量所对应的线段构成的点集为 $PS(V)$,

(3) 有限自动机 M 接受的正则语言为 $L(M)$,

(4) 向量 a 的长度为 $\|a\|$, 串 $w \in L(M)$ 的长度为 $|w|$ 。

定义 1 正则生成系统是一个三元组 $G = (I,$

$P, M)$, 其中:

(1) $I = b_1 b_2 \dots b_m$ 称为初始子;

(2) $P = p \rightarrow q_1 q_2 \dots q_n$ 是向量变换规则或产生式, 满足 $\|p\| > \|q_j\|, j = 1, 2, \dots, n$;

(3) $M = (K, \Sigma, \delta, q_0, F)$ 是一个自动机, 若

$$pT_i = b_i, p\theta_j = q_j, \phi_{ij} = T_i^{-1}\theta_j T_i$$

T_i, θ_j 都是 \mathbb{R}^2 上的相似变换, $i = 1, 2, \dots, m, j = 1, 2, \dots, n$, 则 M 满足:

① $q_0 \in F$,

② $\Sigma = \{\phi_{ij} \mid i = 1, 2, \dots, m, j = 1, 2, \dots, n\}$ 。

图 2 给出了正则生成系统表示例, 图 3 给出了正则生成系统层次组织示意图。

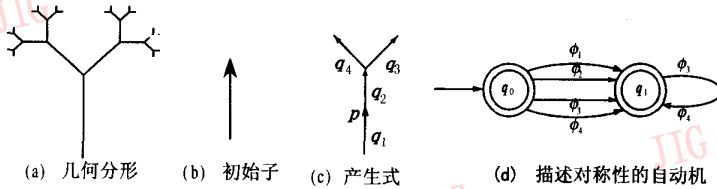


图 2 正则生成系统表示

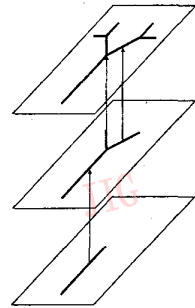


图 3 几何分形的层次表示

定义 2 设 V 是一个向量组, 称 V 是 I 关于正则生成系统 G 的推导, 记为 $I \Rightarrow V$, 当且仅当

$$\forall v \in V \exists b \in I \exists w \in L(M) (v \in IV v = bw)$$

对单个向量也可类似定义。

定义 3 设 $I \Rightarrow V, \lambda \in \Sigma^*$ 是空串且对任意向量 $a\lambda = a$ 。令 $l: V \rightarrow N$ 为

$$l(v) = \max\{|w| : \exists b \exists w (b \in I \wedge w \in L(M) \wedge v = bw)\}$$

则 $I \Rightarrow V$ 的推导长度

$$step(V) = \max\{l(v) : v \in V\}$$

定义 4 设 F 是一个分形, 称正则生成系统 G 是 F 的表示或 G 表示 F 当且仅当

$$\exists V (I \Rightarrow V \wedge F = PS(V))$$

不失一般性, 假设初始子 I 是单个向量, 便有^[2,6]:

引理 1 设正则生成系统 G 中 $P = p \rightarrow q_1 q_2 \dots q_n$ 且 $a \Rightarrow c_1 c_2 \dots c_n$ 。若相似变换 T, θ_j, ξ_j 使得

$$pT = a, p\theta_j = q_j, a\xi_j = c_j$$

则

$$\xi_j = T^{-1}\theta_j T, j = 1, 2, \dots, n$$

引理 2 设正则生成系统 G 中 $I = a, P = p \rightarrow q_1 q_2 \dots q_n, a \Rightarrow c_1 c_2 \dots c_n$ 。又 $pS = a, p\theta_j = q_j, a\phi_j = c_j, j = 1, 2, \dots, n$ 。令

$$T_0 = S, pT_i = a_i, \xi_{ij} = T_{i-1}^{-1}\theta_j T_{i-1}, a_0 = a,$$

$$a_i = a_{i-1}\xi_{ij}, j_i = 1, 2, \dots, n, i = 1, 2, \dots, k$$

则

$$a\xi_{1j_1}\xi_{2j_2}\dots\xi_{kj_k} = a\phi_{i_k}\phi_{j_{k-1}}\dots\phi_{j_1}$$

$$i = 1, 2, \dots, k, j_i = 1, 2, \dots, n$$

有二种描述(准)分形生成过程的方法: 一种是基于 IFS 的变换序列的方法, 一种是基于几何模式推导的方法。引理 2 表明了二者之间的关系: 就生成的点集而言, 变换序列的结果同对应的模式推导序列的逆序列的结果是一致的。

由引理 1、引理 2 和 IFS 的吸引子定理^[1]可得:

定理 1^[6] IFS 的不变集可由正则生成系统表

示。

定理1表明,正则生成系统的表示能力不低于IFS。

2 几何分形的建模

几何分形的建模是指,给定一类几何分形的单个实例,构造出表示它的正则生成系统。由于分形建模在分形几何中称为逆问题,尚没有完全解决^[1],因而本文只对一类几何分形的建模问题进行研究。

令 F 是几何分形, $G = (I, P, M)$ 是 F 的表示且 $P = p \rightarrow q_1 q_2 \dots q_n$ 。记

$$END(a) = \{x \in R^2 : x \text{ 是 } a \text{ 的端点}\}$$

则所研究的一类几何分形 F 满足

- (1) $PS(q_i) \neq PS(q_j), i \neq j, i, j = 1, 2, \dots, n,$
- (2) $PS(q_1 q_2 \dots q_n)$ 是连通的,
- (3) $PS(q_i) \cap PS(q_j) \subseteq END(q_i) \cap END(q_j),$
 $i \neq j, i, j = 1, 2, \dots, n.$

若 $PS(p) \subseteq PS(q_1 q_2 \dots q_n)$, 则称 F 是生长型的, 否则称 F 是膨胀型的。

这一类几何分形可由生成测试法和自动机归纳学习算法构造出正则生成系统的初始子、产生式和自动机。下面首先介绍自动机归纳学习算法, 然后叙述建模方法。

2.1 自动机学习算法

顺序示例集自动机归纳学习算法的特点是, 输入的示例集是严格增序的, 输出的自动机是确定的、最小的。下面给出有关的定义和算法的概要。

定义5 令 $\Sigma = \{\phi_1, \phi_2, \dots, \phi_n\}$ 且 $\phi_i \prec \phi_{i+1}, i = 1, 2, \dots, n - 1,$ 定义序结构 (Σ^*, \prec) :

对所有的 $u, v \in \Sigma^*, u \prec v$ 当且仅当 $|u| < |v|$

或

$|u| = |v|$ 时存在 $\alpha, \beta, \gamma \in \Sigma^*$ 使得 $u = \alpha\phi_i\beta, v = \alpha\phi_j\gamma$ 且 $\phi_i \prec \phi_j$

令 $succ: \Sigma^* \rightarrow \Sigma^*$ 是序结构 (Σ^*, \prec) 元素的直接后继函数, 则顺序示例集是一个有序集:

$$\{\pm \lambda, \pm succ(\lambda), \pm succ^2(\lambda), \dots, \pm succ^m(\lambda)\}$$

其中 $\lambda \in \Sigma^*$ 是空串, 实际示例前面的 \pm 号只出现一个“+”或一个“-”, 分别表示正例和反例。

令 $M^k = (K^k, \Sigma, \delta^k, q_0, F^k)$ 是给定顺序示例集 $\{\pm \lambda, \dots, \pm x\}$ 的学习结果, $minword(q) \in \Sigma^*$ 是 q_0

到 $q \in K^k$ 的永久路径上的串, 算法1给出了进一步给定示例 $\pm w = succ(x)$ 时的学习步骤。

算法1 顺序示例集自动机学习算法

初始化: M^k

输入: $M^k; minword(q_i), q_i \in K^k, i = 1, 2, \dots, |K^k|; \pm w, w = succ(x)$

输出: M^k

过程:

```

old-M ← Mk;
若 w 是正例 accept-w ← true, 否则 accept-w ← false;
new-M ← delete-bad-paths(old-M, w, accept-w);
若 new-M 中没有 w 的路径, 则
repeat
  new-M ← inset-state;
  y ← λ;
  while succ(y) < w {
    y ← succ(y);
    若 old-M 中所有 y 的路径都到达接受状态, 则
      accept ← true;
    否则
      accept ← false;
  }
  new-M ← delete-bad-paths(new-M, y, accept);
until new-M 中存在 w 的路径
输出 new-M

```

初始化自动机 $M^k = (K^k, \Sigma, \delta^k, q_0, F^k)$, 其中 $K^k = \{q_0\}, \Sigma = \{\phi_1, \phi_2, \dots, \phi_n\}, \delta^k(q_0, \phi_i) = q_0, i = 1, 2, \dots, n, F^k = \{q_0\}$ 。算法1应用了二个过程 delete-bad-paths 和 inset-state。delete-bad-paths 删除自动机中的错误路径。inset-state 插入新的状态, 并增加必要的路径。

算法1的细节见文献[7]中的算法, 本文对其中的 delete-bad-paths 和 inset-state 的过程参数作了调整。

2.2 建模方法

给定几何分形 F 后, 建模过程可分为2步。首先应用生成测试法构造出正则生成系统的初始子 I 和产生式规则 P , 同时也构造出从 $I \Rightarrow F$ 的所有中间向量, 即生成迹。然后根据生成迹生成学习示例, 并应用自动机归纳学习算法构造出自动机 M 。下面给出生成迹的定义及主要算法。

定义6 令 F 是几何分形, $G = (I, P, M)$ 是 F

的表示, $I \Rightarrow V$ 且 $PS(V) = F$ 。则, 集合

$$D = \{v \mid v \in U \wedge I \Rightarrow U \wedge \text{step}(U) \leq \text{step}(V)\}$$

称为 F 关于 G 的生成迹。

生成迹是构造自动机归纳学习示例的基础。几何分形与其生成迹是不同的, 见图4。

算法2 几何模式构造算法

输入: 几何分形 F

输出: 生成迹 D , 初始子 I , 产生式 P

过程:

```

{
  distance = 2; success ← false;
  e ← F 中最短的线段; pattern ← {e};
  E ← {x ∈ F | x 可达 e 且可达路径长度小于等于 distance}
  while E ≠ ∅ ∧ not success {
    f ← E 中与 pattern 关联的最短线段;
    E ← E - {f}; pattern ← pattern ∪ {f}; head ← ∅; h ← e;
    while h ∉ head {
      head ← head ∪ {h}; h ← Phead(pattern);
      direction ← 0;
      while direction < 2|pattern|+1 ∧ not success {
        按 direction 为 pattern 中的线段和 h 定向; D ← ∅;
        repeat
          d ← Reduce(h, pattern, F, D); D ← D ∪ d;
        until d = ∅;
      }
    }
  }
}
    
```

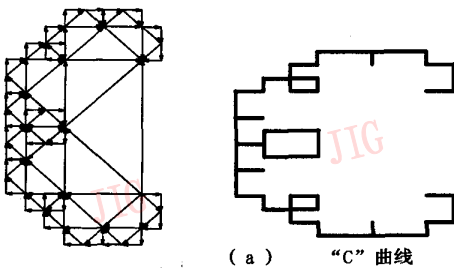


图4 “C”曲线生成迹

在构造出初始子 I 和产生式 P 后可计算出自动机 M 的字母表 $\Sigma = \{\phi_1, \phi_2, \dots, \phi_n\}$, 再利用生成迹 D 可构造出顺序示例集, 进而构造出正则生成系统的自动机 M 。

令 $I = a, P = p \rightarrow q_1 q_2 \dots q_n, a \leq b_1 b_2 \dots b_n$ 。 $pT = a, p\theta_i = q_i, a\phi_i = b_i, T, \theta_i, \phi_i$ 是 \mathbb{R}^2 上的相似变换, $i = 1, 2, \dots, n$ 。则由引理1

若 $F - PS(D) = \emptyset$ 则 $\text{success} \leftarrow \text{true}$;

$\text{direction} \leftarrow \text{direction} + 1$;

若 $\text{success} = \text{true}$ 则

$P \leftarrow \text{Simplify}(\text{pattern}, h, \text{direction})$;

$D \leftarrow \text{Refine}(D)$; $I \leftarrow D$ 中最长的向量;

输出 (D, I, P) ;

否则

输出 (“建模失败!”);

算法2作用的是生成测试法, 主要过程有模式归约过程 Phead、几何分形归约过程 Reduce、生成迹精化过程 Refine 和产生式简化过程 Simplify。Phead 过程构造一个没有定向的产生式 $h \rightarrow \text{pattern}$ 。若 $h \in \text{pattern}$, 则在 pattern 中加入 h 的两个二分线段。Reduce 过程按照定向的产生式 $h \rightarrow \text{pattern}$ 进一步归约 F (已归约的不再归约), 并对由 h 定向的向量记录 Reduce 过程成功的次数。Refine 过程删除 D 中除最长向量之外计数为 1 的向量及其一阶推导, 若产生式被简化过, 则在 D 中再加入一些中间推导向量。Simplify 过程对产生式进行必要的简化。distance 是一个常数, 用来限定产生式的搜索范围。算法2不破坏 F , 这是必要的, 因 F 中的某线段可能是多个向量重合的结果且可定向为不同的向量。见图4和图5。

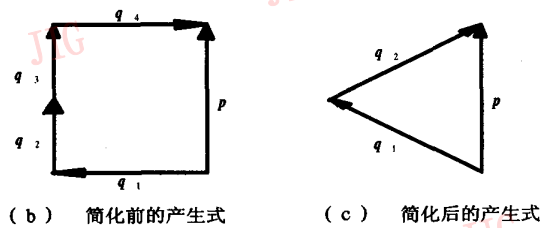


图5

$$\phi_i = T^{-1}\theta_i T, \quad i = 1, 2, \dots, n$$

算法3 自动机构造算法

输入: 初始子 a , 字母表 $\Sigma = \{\phi_1, \phi_2, \dots, \phi_n\}$, 生成迹 D

输出: 自动机 M

过程:

$\text{depth} = 3; M \leftarrow M^t$;

$w \leftarrow \lambda; lc \leftarrow 0$;

```

while  $lc < |\Sigma|^{depth}$  {
   $w \leftarrow succ(w)$ ;
   $aw \in D$  则  $+w$  否则  $-w$ ;
   $M \leftarrow M$  按算法 1 学习示例  $w$  的结果;
   $lc \leftarrow lc + 1$ ;
}

```

3 实验例子与分析

对文献中的若干几何分形(包括 Koch 曲线、“C”曲线和树形植物等^[2-5])进行了建模实验。图 6 和图 7 给出了其中一个几何分形建模过程中的正例集和自动机学习步骤,该分形是文献[4]中应用 LRIFS 表示的一个分形。在自动机构造过程中, $\pm \phi_i \phi_j \dots$

ϕ_k 简记为 $\pm ij \dots k$, 从几何分形的生成迹(有向边集合)中可生成示例集,如下所示:

- $\{ +\lambda, +1, +2, +3, +4, -11, -12, +13, +14,$
- $-21, -22, +23, +24, -31, 32, +33, +34, -41,$
- $-42, +43, +44, -111, -112, -113, -114,$
- $-121, -122, -123, 124, -131, -132, +133,$
- $+134, -141, -142, +143, +144, -211, -212,$
- $-213, -214, -221, -222, -223, -224, -231,$
- $-232, +233, +234, 241, -242, +243, +244,$
- $-311, -312, -313, 314, -321, -322, -323,$
- $-324, -331, -332, +333, +334, -341, -342,$
- $+343, +344, -411, -412, -413, -414, -421,$
- $-422, -423, -424, -431, -432, +433, +434,$
- $-441, +442, +443, +444 \}$

建模过程的结果如图 2 所示。

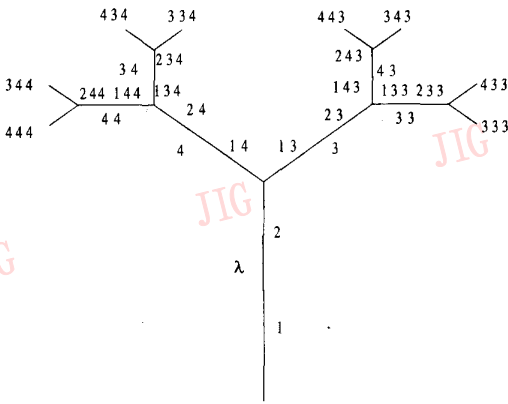


图 6 自动机归纳学习的正例

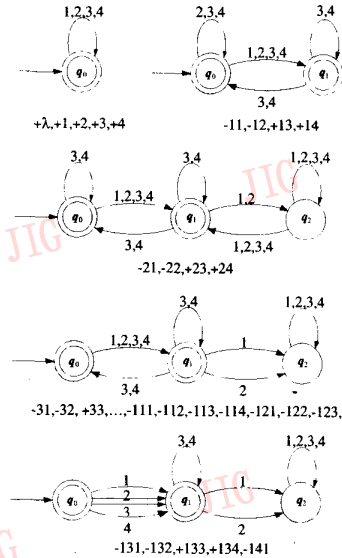


图 7 自动机归纳学习步骤

正则生成系统与 LRIFS 的不同之处是:(1) 正则生成系统的形状基元是向量(有向线段),而 LRIFS 的形状基元是点。由 IFS 吸引子定理可知, LRIFS 只能表示生长型的几何分形,不能表示膨胀型的几何分形^[1,6];而正则生成系统两者都能表示。(2) 在描述对称性方面,正则生成系统和 LRIFS 都使用了有限自动机,但正则生成系统比 LRIFS 简单,见图 8。(3) 正则生成系统是一种拟真形状表示,由于初始子、产生式和有限自动机有高效的比较算法,

因而正则生成系统有高效的形状识别和匹配算法^[6]。

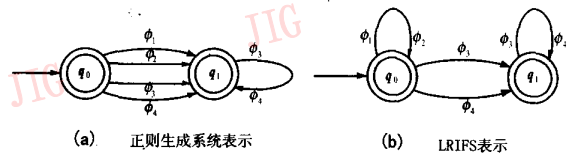


图 8 正则生成系统与 LRIFS 的比较

4 结束语

本文在 Prusinkiewicz 和 Gujar 等人工作的基础上,提出了几何分形的正则生成系统表示模型,并进一步研究了一类几何分形的建模问题。实验结果表明,正则生成系统不仅能够给出几何分形简单有效的表示,而且便于进一步的分析和处理。

本文的实验对象是理想的几何分形。进一步的工作是研究自然物体的正则生成系统表示和建模问题。

参考文献

1 肯尼思·法尔科内著,曾文曲,刘世耀,戴连贵等译.分形几何

——数学基础及其应用.沈阳:东北大学出版社,1993.

- 2 Prusinkiewicz P, Sandness G. Koch curves as attractors and repellers. IEEE Computer Graphics & Application, 1988, 8(6):26~40.
- 3 Gujar U G, Bhavsar V C. Traversed geometric fractals. IEEE Computer Graphics & Application, 1993, 13(5):61~67.
- 4 Prusinkiewicz P, Hammel M S. Escape-time visualization method for language-restricted iterated function systems. In: Encarnacao J L, Peitgen H-O, Sacks G *et al* (eds). Fractal Geometry and Computer Graphics. Berlin:Springer Verlag, 1992, 24~44.
- 5 Holliday D J, Samal A. Object recognition using L-system fractals. Pattern Recognition Letters, 1995, 16(1):33~42.
- 6 廖士中. 定性空间推理分层逼近方法研究:[学位论文].北京:清华大学计算机科学与技术系,1997.
- 7 Porat S, Feldman J A. Learning automata from ordered examples. Machine Learning, 1991, 7(2):109~138.



廖士中 博士,副教授。主要研究方向为图象分析,空间推理和人工智能。



石纯一 教授,博士生导师,1957年毕业于北京大学数学力学系数学专业。现从事人工智能应用基础研究,发表论文百余篇,译著11本,完成国家课题近20项。

A Model for a Class of Geometric Fractals

Liao Shizhong

(Department of Computer Science, Liaoning Normal University, Dalian 116029)

Shi Chunyi

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

Abstract In this paper, on the basis of Iterated Function Systems and Finite Automaton theories, a geometric fractals representation model named regular generation system is proposed. Then, on the basis of generation and test, a modeling algorithm of regular generation system is constructed by using automata inductive learning method. Finally, the experimental result is shown to demonstrate that the model is simple and valid.

Keywords Geometric fractal, Fractal modeling, Automata inductive learning, Shape representation