Syntactic Interpolation of Fractal Sequences

Jacques Blanc-Talon

CTA/GIP
16 bis, Avenue Prieur de la côte d'or,
94114, Arcueil, FRANCE
blanc@etca.fr

Abstract A grammatical inference algorithm is described which computes a context-free grammar interpolating fractal encodings. The "fractality" of the initial curve is captured by the symbolic part of the algorithm while the approximation is performed by the numerical part. Hybridization of the algorithm consists in exchanging some information between the two parts, i.e. by translating the eigenvalue variations of the growth matrix into syntactic variations while checking the compatibility with the symbolic system already inferred.

1 Introduction

If a lot of works have been published on pattern recognition in the previous decades, there are few available on the recognition of fractal sets. Fractals are easy to generate and yield nice art displays but once the wonder has vanished the fascinating inverse problem of computing the models from real data remains, which is still unviolated. Even with models as simple as IFSs [12] the inverse problem may lead to untractable computations; it is thus impossible to compute more advanced models, even if desirable for the sake of generalization and for practical reasons. Nevertheless, the key to unlock fractal models seems to be the relationship between their internal structure and the basic elements they are made of.

Structural Pattern Recognition (SPR) gathers a wide set of techniques whose distinctive feature is to give major importance to some special information against the set of patterns to process. In fact, recognition is achieved by combining information of two different natures: firstly some disparate information consisting in a decomposition over a set of primitive patterns and called the *alphabet* and secondly some long-range information describing relationships between subsets of patterns called the *structure*. Accepted patterns must present both alphabetical and structural information compatible with the model but most of the time, the two are straightforwardly juxtaposed since they do not belong to the same physical or mathematical space.

In this paper, a method for computing models from "fractal encodings" of sets is discussed. Given a sequence of encodings, the underlying structure of the model is computed first, then augmented with disparate information. In order to make the first and the second one compatible, we use a syntactic model to provide a unified framework for dealing with such information at various semantical levels. This model is a Context-Free Grammar (CFG) because CFGs are appropriate to fractal modelling[1] (they allow intricate recursive derivations) and because any "non fractal" noise may be modelled by a regular language (which does not change the overall structure [7]). The syntactic optimization runs together with a numerical algorithm so as to *infer* the grammatical model.

This method has been tested on encodings of 2D curves generated by just-touching IFSs. However, the practical application which gave birth to this study was far more complex: it dealt with the modeling of gold ore distribution in a mining area [11]. Because such a distribution exhibits fractal features, syntactic techniques seemed suitable for modeling huge data sets.

The paper is organized as follows. Theoretical background is recalled in Sect. 2. An encoding process which extracts syntactic and structural information simultaneously is discussed in Sect. 3. Sections 4 and 5 detail the very core of the inference method. In the conclusion, we emphasize the use of syntactic models for processing fractal information.

2 Preliminaries

L-systems are rewriting machines which can be used for generating self-affine fractal curves [5, 8]. For instance, Fig. 1 shows the attractor of a D0L-system whose rules once concatenated yield the sentence "SSPR should take place in Australia", the initial word being "Sydney". D0Ls [14] are among the simplest L-systems. Although "innocent-looking", they allow the computation of sequences with rather complex combinatorial properties. Moreover, the diversity of curves gets wider if several morphisms are used.

In this paper, T is a finite set and T^* the free monoid of words; 1 is the null letter. For $t \in T, x \in T^*$, $|x|$ denotes the length of x and $|x|_t$ the number of occurrences of t in x. Also, $x[k]$ is the subword made of the k first letters of x. A D0L is a triple (T, h, s) which generates the language $\mathcal{L}(D) = \{h^n(s) = h \circ h^{n-1}(s), n \geq 0\}$.

$d_L(x, y)$ denotes the Levenstein distance (sometimes called *edition distance*) between strings $x, y \in T^*$. This distance is defined very intuitively by means of three basic operations, *del* (-etion), *ins* (-ertion) and *subs* (-titution):

$$
\begin{aligned}
& del(1) = \{1\} \\
& \forall\, a \in T,\ del(a) = \{1\} \\
& \forall\, u, v \in T^* \setminus \{1\},\ del(uv) = del(u)\, v + u\, del(v) \\
& ins(1) = T \\
& \forall\, a \in T,\ ins(a) = T\, a + a\, T \\
& \forall\, u, v \in T^* \setminus \{1\},\ ins(uv) = ins(u)\, v + u\, ins(v)
\end{aligned}
$$

[1] Let us mention the growing interest about grammars for fractal generation [9, 10, 13].

$$subs(1) = \{1\}$$
$$\forall\, a \in T,\ subs(a) = T \setminus \{a\}$$
$$\forall\, u, v \in T^* \setminus \{1\},\ subs(uv) = subs(u)\, v + u\, subs(v)$$

Let x be a string, and $(C_n(x))_{n\in\mathbb{N}}$ the balls built recursively according to:

$$C_0(x) = \{x\}$$
$$C_{n,n\geq 1}(x) = del\,(C_{n-1}(x)) + ins\,(C_{n-1}(x)) + subs\,(C_{n-1}(x))$$

Then the *Levenstein distance* between x and y is:

$$d_L(x, y) = \inf\{n \in \mathbb{N} |\ y \in C_n(x)\}$$

Let us recall a few results about the perturbation of the roots of a polynomial and the eigenvalues of a matrix (see [15] for instance). In the following, vectors are in bold font; the entries of any matrix A lie in brackets: $(a_{i,j})$, its transpose is A^T, $|A|$ is the matrix whose elements are $|a_{i,j}|$. I_n is the identity matrix of size $n \times n$. $\sigma(A)$ is the spectrum of A. $\delta_{i,j}$ is the Kronecker delta.

$||.||$ denotes either the usual Euclidean vector norm or the Frobenius matrix norm; the latter has been chosen since it satisfies the triangle inequality for matrix multiplication.

We shall use the following proposition and theorems:

Proposition 1. *Let $A \in \mathcal{M}_n(\mathbb{R}), \mathbf{b}, \mathbf{c} \in \mathbb{R}^n$ then (formally) :*

$$(A + \mathbf{c}\mathbf{b}^T)^{-1} = A^{-1}\left(I_n - \frac{\mathbf{c}\mathbf{b}^T A^{-1}}{1 + \mathbf{b}^T A^{-1}\mathbf{c}}\right)$$

Theorem 1 (Bauer-Skeel). *Let A be nonsingular; let $A\mathbf{x} = \mathbf{b}$ and $(A+E)\tilde{\mathbf{x}} = \mathbf{b} + \mathbf{e}$. If for some nonnegative $S, \mathbf{s}$ and ϵ: $|E| < \epsilon S$ and $|\mathbf{e}| < \epsilon \mathbf{s}$ and in addition $\epsilon\, ||\, |A^{-1}|\, S|| < 1$ then*

$$||\tilde{\mathbf{x}} - \mathbf{x}|| \leq \frac{\epsilon\, ||\, |A^{-1}|\, (S\, |\mathbf{x}| + \mathbf{s})||}{1 - \epsilon\, ||\, |A^{-1}|\, S||}$$

Many theorems yield inequalities for the roots of perturbed polynomials; the following recent one refines Ostrowski's. Let the Bombieri's norm of a polynomial $P = \sum_{i=1}^{n} a_i x^i$ be:

$$[P]_B = \sqrt{\sum_{i=0}^{n} \frac{i!(n-i)!}{n!}\, |a_i|^2}$$

Theorem 2 (Beauzamy [4]). *Let P and Q be two polynomials defined as: $P(x) = \prod_{i=1}^{n}(x - x_i)$ and $Q(x) = \prod_{i=1}^{n}(x - y_i)$ and such that $[P - Q]_B < \epsilon$. If all the roots of P are distinct and if*

$$\epsilon \leq \frac{1}{2n} \min_j \frac{|P'(x_j)|}{(1 + |x_j|^2)^{(n-1)/2}}$$

then for every j a root y_i of Q is such that

$$|x_j - y_i| < 2n\frac{(1 + |x_j|^2)^{n/2}}{|P'(x_j)|}\epsilon$$

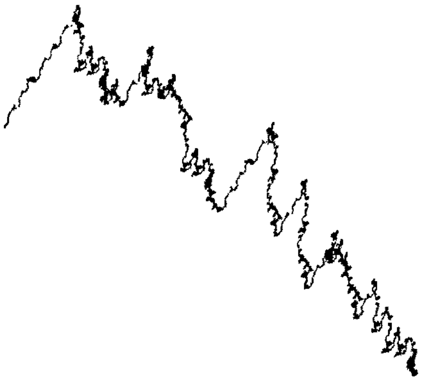

Figure1. Iterating the rules "SSPR should take place in Australia" on axiom "Sydney" (41,357 symbols)

3 Encodings of Real Sets

The general idea of fractal coding is given in [3]. We have introduced in previous papers [5,7] an image encoding method based on a recursive segmentation of 2D sets. This method yields non-balanced quadtrees [1] whose terminal leaves are subimages taken within the original image. Quadtree encoding is easy to implement, quadtrees meet accuracy issues since the general quadtree is but the image itself and mapping any quadtree to a screen can be performed very efficiently. In the present case, the whole image is assumed to be a (possibly infinite) collection of self-affine copies of a few source subimages.

The comparison between images is performed by an enhanced correlation operator which is able to manage rotation and scaling of the sets to be compared. The leaves are the terminal letters of the final grammar while the branching nodes of the quadtree are considered as non-terminal symbols. They are rewritten as a word of the nodes (terminal or not) they are giving access to. Figure 2 shows a self-affine curve; rectangular boxes are terminal leaves linked by correlation maximization.

This method has given satisfactory results for self-affine sets but unfortunately it cannot be applied to multifractal sets which are anything but self-affine! Thus, the algorithm discussed below makes no use of any extra structural information delivered with the encodings. Instead, it only makes the assumption that the encodings reflect the (multi-) "fractality" of some initial phenomenon one wants to model. In the formal languages terminology, "fractality" implies "iteration" and requires at least the use of context-free grammars [7].

Nevertheless, it is always possible to encode a Jordan curve as a sequence of words highlighting some iteration process (provided there is some theoretical justification in doing so !) as pointed out in [6]. For instance, the number of symbols $N(\epsilon)$ for encoding a fractal curve as a function of a small line segment behaves as $A\epsilon^{-D}$ where D is the fractal dimension; various algorithms are available for computing D and for performing Freeman encoding.

As the curve is assumed to be interpolated by a non-erasing D0L-system, Perron-Frobenius' theorem yields $N(\epsilon) \sim B(\lambda_{\max})^k$ for some k, $\lambda_{\max}$ being the (positive) greatest eigenvalue of the growth matrix (see Sect. 4). Since ϵ is a parameter of the encoding algorithm (ϵ may be a function of the encoding accuracy), it is possible to plot $(\log(\epsilon), log(N(\epsilon)))$ so as to find $\lambda_{\max}$; the best integer sequence $k_1, k_2, \ldots$, and the related encodings are computed accordingly. We performed such a process successfully on just-touching IFS curves [5].

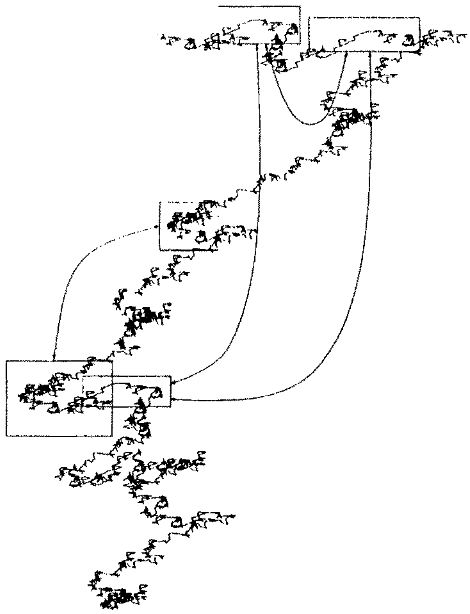

Figure2. Possible correlation locations in a self-affine curve

4 D0L Computation and Grammatical Inference

As an input, let us consider a sequence $S = \{x_1, x_2, \cdots, x_p\}$ of words over $T = \{t_1, t_2, \cdots, t_n\}$ (with $n+1 < p$) and the related sequence of integer vectors: $g_k = [|x_k|_{t_1}, |x_k|_{t_2}, \cdots, |x_k|_{t_n}]$, $1 \leq k \leq p$. S can be mapped to $\mathbb{R}^n$ by using an expansive morphism $K : T^\star \to \mathbb{R}^n$ which is not needed to be detailed here. The "fractal space" is the complete space $(\mathbb{R}^n, h)$ where h is the Hausdorff metric. One assumes there is an approximation of S by a regular D0L-sequence, i.e.

$$x_k = \mu_k \circ \phi^k(\tilde{x}_0),\ 1 \leq k \leq p\ , \tag{1}$$

with $\phi : T^\star \to T^\star$ a non-erasing morphism, $\tilde{x}_0$ some initial word[2] and μ_k a transduction such that

[2] Please note that symbol "$\sim$" means that the adressed value is the one related to the optimal case, i.e. without noise.

$$d_L(\mu_k \circ \phi^k(\tilde{x}_0), \phi^k(\tilde{x}_0)) < q, 1 \leq k \leq p \ . \tag{2}$$

Previous equation can be rewritten in the numerical space as

$$\mathrm{g}_k = M^k \tilde{\mathrm{g}}_0 + \theta_k, \ 1 \leq k \leq p \ . \tag{3}$$

In this equation $M \in \mathcal{M}_n(\mathbb{N}), \mathrm{g}_k \in \mathbb{N}^n$, $\tilde{\mathrm{g}}_0$ is an initial vector and θ_k stands for some bounded noise ($\sup_{1 \leq k \leq p} |\theta_k| = q < \infty$) whose distribution is assumed to be uniform over T. One must notice that this condition is rather general: the noise may be *anything* but a cascading process. In general, $\sup_k |\theta_k| \leq \sup_l d_L(\mu_l \circ \phi^l(x_0), \phi^l(x_0))$. It is important that q is finite as stated by:

Proposition 2 (Blanc-Talon [7]). *If (u^n) is an increasing sequence (according to $\leq$) whose limit is u, and if (v^n) is a sequence such that $\exists\, p, m \in \mathbb{N}, \forall\, n > m, d_L(u^n, v^n) \leq p$, then their mappings $K(u^n)$ and $K(v^n)$ in $\mathbb{R}^n$ converge to the same curve in the Hausdorff metric h.*

M is the growth matrix of morphism ϕ and its entries are $|\phi(t_i)|_{t_j}$. Inferring the D0L-system in the perfect case ($\forall k, \theta_k = 0$ and $q = 0$) can be achieved either in the symbolic space or in the numerical space. However, neither method works in case of noisy sequences. The new following approach consists in using the partial results computed by a numerical algorithm as the input of a symbolic method and vice-versa, as long as the result after the current pass is not correct. The maximum number of loops is n $(= \#T)$.

First, one determines the characteristic polynomial $\tilde{P}$ of the perturbed matrix $\tilde{M}$, written R for sake of clarity, from the set of vectors g_k. Since the noise has an upper limit, the method converges to the correct values: the coefficients of $\tilde{P}$ have continuous variations according to the noise, and so do the roots if this noise is small enough. Then, perturbed eigenvalues are computed, which are located in some Gerschgorin disks of the exact values we are looking for. We use a refined estimate of the disc for computing their variation range and possible integral matrices are constrained by this result. The symbolical part consists in determining the morphism rules with the greatest row sum; it is performed by considering the number of occurrences of subwords of increasing length.

Please notice that the proofs are given below in the scalar case which lightens the notations (a lot !). Instead of considering the sequence of vectors g_k, one considers a sequence of vector entries ($g_1 = \mathrm{g}_1(l), \cdots, g_p = \mathrm{g}_p(l)$). This sequence also forms a vector; let g denote this vector. The following formulas are still valid for vectors, however in a more complex form. Given $\#T$, there is indeed a particular l for which the whole method does work (i.e. matrix G is nonsingular). Approximating $\tilde{P}$ by P implies computing the best hyperplane through $g_1, \cdots, g_p$:

$$\frac{\partial}{\partial a_m} \sum_{k=n+1}^{p} \left(g_k - \sum_{l=1}^{n} a_l g_{k-l} \right)^2 = 0 \ , \tag{4}$$

which yields:

$$\mathbf{a} = G^{-1}\mathbf{u} = [(sg(i,j))_{i,j=1}^{n}]^{-1}(sg(i,0))_{i=1}^{n} \tag{5}$$

with $G \in \mathcal{M}_n(\mathbb{H}), \mathbf{u} \in \mathbb{H}^n$ and the entries being $sg(i,j) = \sum_{k=n+1}^{p} g_{k-i}g_{k-j}$. Solutions of

$$x^n - \mathbf{a}^T(x^k)_{k=0}^{n-1} = x^n - \sum_{k=0}^{n-1} a_k x^{n-k} = 0 \tag{6}$$

are the estimated eigenvalues. One has $\tilde{G} = G - (\theta\theta^T + \mathbf{g}\theta^T + \theta\mathbf{g}^T)$. Perturbation of $\tilde{G}^{-1}$ can thus be computed by setting $A = G - (\theta\theta^T + \mathbf{g}\theta^T), b = \mathbf{g}, c = -\theta$ in (1) which allows Theorem 1 to yield the max of $||\mathbf{a} - \tilde{\mathbf{a}}||$, that is the perturbation over polynomial coefficients. The next step is to quantify the spectrum variation: assuming that the roots of P are distinct Theorem 2 applies :

$$|\lambda_i - \tilde{\lambda}_j| \leq 2n\,|\lambda_i|\,\frac{(1+|\lambda_i|^2)^{(n/2)}}{\left|\sum_{k=1}^{n} k a_k \lambda_i{}^{n-k}\right|}\,||\mathbf{a} - \tilde{\mathbf{a}}|| \quad . \tag{7}$$

This result states that one can deduce the range of variations of $\sigma(R)$ with respect to $\sigma(M)$ from the bounds of θ_k. How is it possible to relate $||R - M||$ itself to the former ? Since $\tilde{P}(\lambda) = Det(M - I_n\lambda)$, taking derivates with respect to any $\tilde{a}_m$ on both sides yields:

$$\lambda^m + m\tilde{a}_m\lambda^{m-1}\frac{\partial\lambda}{\partial\tilde{a}_m} = \sum_{k=1}^{n} Det(Q(k)) \ , \tag{8}$$

$$\text{with } Q(k) = \begin{cases} m_{i,j} - \delta_{i,j}\lambda & i \neq k \\ \dfrac{\partial m_{i,j}}{\partial\tilde{a}_m} - \delta_{i,j}\dfrac{\partial\lambda}{\partial\tilde{a}_m} & i = k \end{cases} \ . \tag{9}$$

This expansion in terms of determinants of $Q(k)$ *is very important* since every line of R reflects the related rule $\phi(t_k)$. Taking $\tilde{\lambda}$ in the neighborhood of every λ_i allows us to determine the upper bound of the entries of R as a function of the noise; the lower bound is always zero since ϕ is assumed to be non-erasing. Since $M \in \mathcal{M}(\mathbb{N})$, the problem can be solved by a minimization algorithm around initial entries of R. Simulated annealing has been used for finding M which minimizes $||\mathbf{g}(m+1) - M\mathbf{g}(m)||$ under the constraint $0 \leq m_{i,j} \leq$ [variation given by (8)].

Thus, the whole algorithm can be summarized as follows.

1. Compute $\sigma(R)$ from $g(1), g(2), \cdots, g(p)$.
2. Compute the possible variations of $r_{i,j}$ from values $a_1, a_2, \cdots, a_n$.
3. Compute the rewrite rules from the longest and the most frequent subwords in S.
4. Fill matrix R and perform the minimization described above.
5. If the variations of M are greater than those allowed by (8), choose the next possible subwords and repeat Step 4.

5 A CFG and a Regular Transduction

A CFG can be derived from the D0L computed above very easily. Briefly, every terminal letter in every rule $\phi(t) = s, t \in T, s \in T^*$ is replaced by a new non-terminal symbol and by adding the related "terminal rules". Given $D = (T, \phi, s)$, let us consider an auxiliary set N and a one-to-one mapping $\mu : T \to N$

Definition 1.

$$G_D = (T, N, \mu(s), \{\mu(t) \to \mu(\phi(t)), t \in T\} \cup \{\mu(t) \to t, t \in T\})$$

Moreover, one has to add significant rules for changing these "perfect" words into the real ones. Practical results showed that it is unwise to add these rules directly to the grammar: the diversity of curves accepted by a syntactic parser gets too wide. Thus, we decided to model the difference between the context-free language and the real samples by a transduction γ_S, that is a subset $T^- \times T^+$ of $T^* \times T^*$. Let t^- (respectively t^+) denote the cardinal of T^- (resp. T^+). Since the noise is bounded, γ is a regular transduction ([2], an absolute reference).

Inference of γ_S is achieved by computing the sequence of distances between the "perfect" words $\phi(x_k), 0 < k < p$ (x_0 is unknown) and the real samples x_{k+1}. The algorithm for computing $d_L(\tilde{x}_{k+1}, x_k)$ yields the trace of the distance, that is, the minimum set of basic operations used to transform any of the two words into the other one. Such a trace defines a set of local transductions $\Gamma(k) = \{\gamma_k^1, \gamma_k^2, \cdots, \gamma_k^{p_k}\}$ for every k; let Γ_S defines the union of local transductions overall the set S of words: $\Gamma_S = \bigcup_{0<k<p} \Gamma(k)$.

The (finite) transduction γ_S is the union for every k of the local transduction $\gamma_k = T_k^- \times T_k^+$ minimizing $\sum_{1<k<p} t_k^-$ and $\sum_{1<k<p} t_k^+$ simultaneously.

6 Concluding Remarks

Syntactic recognition of fractal patterns turns out to be a new promising field of research and applications. Promising, first because it shares a common background with usual syntactic pattern recognition: one has to dig into formal languages theory in order to find a strong theoretical framework. In fact, the difference lies only in the nature of the model which is assumed to be fractal. But this point is actually essential, for syntactic models *can* encode this fractality explicitly. Secondly, this fractality is the key to establishing a fruitful connection between both theories of formal languages and dynamical systems. Fascinating problems such as the modification of geometrical generative power of grammars as a function of the mapping appear at the interface of this connection.

Syntactic techniques exhibit real advantages over numerical and statistical techniques (despite [16], a very pessimistic paper !) in the fractal case. Fractality, considered as a recursion scheme, is naturally encoded by means of CFGs. High-level fractal information is processed at the same level as the disparate information.

References

1. Abdallah, A.N., Berstel, J.: Tétraarbres engendrés par des automates finis, LITP, Université Paris 6, France (1989)
2. Autebert, J.-M., Boasson, L.: Transductions rationnelles, Masson (1988)
3. Barnsley, M.: Fractals everywhere, Academic Press (1988)
4. Beauzamy, B.: How the roots of a polynomial vary with its coefficients : A local quantitative result, Bull. Canadien de Mathématiques (1998)
5. Blanc-Talon, J.: How to fill the gap between Fractals and Stochastic Context-Free Grammars?, J. Complexity International (1996)
6. Blanc-Talon, J.: MultiFractal Grammars for more Complexity, NOLTA'95, "Non-Linear Theory and its Applications" (1995)
7. Blanc-Talon, J.: Generation and Recognition of Fractal sets by using a syntactic approach, J. Complexity International (1994)
8. Dekking, F.M.: Recurrent sets, J. Advances in Mathematics, 78–104, **44** (1982)
9. Fernau, H.: IFS and codes, 7th Int. Meeting of Young Computer Scientists (1992)
10. Bordihn, H.: On the degree of nondeterminism of grammars, 7th Int. Meeting of Young Computer Scientists (1992)
11. Kravis, S.P., Veldkamp, P., Horowitz, F.G.: Rendering IFS Encoded 3D Objects on a MasPar MP-1, Tech. Report **22**, CSIRO-DIT, Canberra, Australia, (1992)
12. Massopust, P.R.: Fractal Functions, Fractal Surfaces, and Wavelets, Academic Press (1994)
13. Nolle, M.: Comparison of different methods for generating fractals, 7th Int. Meeting of Young Computer Scientists (1992),
14. Rozenberg, G., Salomaa, A.: The Mathematical Theory of L-systems, Academic Press (1970)
15. Stewart, G.W., Sun, J.-G.: Matrix Perturbation Theory, Academic Press (1990)
16. Tanaka, E.: Theoretical Aspects of Syntactic Pattern Recognition, J. Pattern Recognition, **7** (1994)