Fractal Inverse Problem: Approximation Formulation and Differential Methods
Eric Guérin, Eric Tosan

To cite this version:

HAL Id: hal-01537632
https://hal.archives-ouvertes.fr/hal-01537632
Submitted on 12 Jun 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Fractal Inverse Problem: Approximation Formulation and Differential Methods

Éric Guérin\(^1\) and Éric Tosan\(^2\)

\(^1\) LIRIS - Université Claude Bernard - Batiment Nautibus - 43, Bd du 11 Novembre - 69622 Villeurbanne Cedex - France - eric.guerin@liris.cnrs.fr

\(^2\) LIRIS - Université Claude Bernard - Batiment Nautibus - 43, Bd du 11 Novembre - 69622 Villeurbanne Cedex - France - eric.tosan@liris.cnrs.fr

1 Introduction

1.1 Fractal Inverse Problem

The fractal inverse problem is an important research area with a great number of potential application fields. It consists in finding a fractal model or code that generates a given object. This concept has been introduced by Barnsley with the well known collage theorem [2]. When the considered object is an image, we often speak about fractal image compression. A method has been proposed by Jacquin to solve this kind of inverse problem [13].

This problem has been studied by many authors. Generally speaking, inverse methods can be classified in two types:

- Direct methods: model characteristics are found directly. In the fractal case, very few direct methods have been proposed. In general, we have to deal with synthetic data entries. Some authors use wavelet decomposition to find frequency structures and extract IFS coefficients [3, 16]. A method using complex moment has been experimented to work for fractal images [1].
- Indirect methods: model characteristics are found indirectly. In general an optimization algorithm is used. These methods allow to deal with more complex models and less synthetic data entries. Inverse problem for mixed IFS has been performed with genetic methods [14].

Optimization methods used in indirect methods are generally stochastic, because it’s not possible to calculate any derivative with respect to the model parameters.

1.2 Fractal approximation

In [17], Vrscay and Saupe introduced a derivative property in the numerical functions involved in fractal image coding with respect to their affine IFS
parameters. This allows the use of a gradient descent method to solve the inverse problem and gives better results than the standard collage theorem.

In [8, 9], we developed a method based on this property for fractal approximation of curves and surfaces. In [10, 11], we have extended this method to surfaces. In [12, 6], we introduced a projected IFS tree model to obtain a better approximation of natural surfaces and greyscale images.

In this paper we develop a differential approach of the fractal approximation problem based on formal multiresolution.

1.3 Formal multiresolution

In [7], we present an analytical approach of the approximation problem based on address functions. Barnsley introduces these functions to define a formal parameterization of attractors [2]. Address functions are functions that takes address arguments rather than numerical ones. They map from infinite words $\Sigma^\omega$ to a modelisation space $\mathcal{X} = \mathbb{R}^m$:

$$\phi : \Sigma^\omega \rightarrow \mathcal{X}$$

$$\rho \mapsto \phi(\rho)$$

Address functions give a natural multiresolution formulation [7]: by selecting infinite words of $\Sigma$ that have the form $\alpha k^\omega$, we define a finite family of points that can be viewed as the tabulation of $\phi$ at a given level $n$:

$$(\phi(\alpha k^\omega))_{\alpha \in \Sigma^n, k \in \Omega}$$

where $\Omega$ is a subset of $\Sigma$ such that $\phi(k^\omega)$ represents the "boundaries" of the figure.

The modelisation space $\mathcal{X} = \mathbb{R}^m$ is Hilbertian, we define a new Hilbert space of address functions based on the following dot product[7]:

$$\langle \phi, \phi' \rangle = \lim_{n \to \infty} \frac{1}{N^n} \sum_{\alpha \in \Sigma^n} \frac{1}{M} \sum_{k \in \Omega} \langle \phi(\alpha k^\omega), \phi'(\alpha k^\omega) \rangle$$

with $M = |\Omega|$ and $N = |\Sigma|$. This constitutes a general frame of geometric fractal modeling and approximation.

2 IFS approximation

One convenient way to provide address functions is the use of IFS (Iterated Function Systems). Furthermore, we will see that these functions verify a decreasing condition and then belong to $L^2(\Sigma^\omega, \mathcal{X})$. 
2.1 IFS model

Introduced by Barnsley [2] in 1988, the IFS (Iterated Function Systems) model generates a geometrical shape or an image [13] with an iterative process. An IFS-based modeling system is defined by a triple \((\mathcal{X}, d, S)\) where [18, 19]:

- \((\mathcal{X}, d)\) is a complete metric space, \(\mathcal{X}\) is called iteration space;
- \(S\) is a semigroup acting on points of \(\mathcal{X}\) such that: \(\lambda \in \mathcal{X} \mapsto T\lambda \in \mathcal{X}\) where \(T\) is a contractive operator, \(S\) is called iteration semigroup.

An IFS \(T\) (Iterative Function System) is a finite subset of \(S\): \(T = \{T_0, ..., T_{N-1}\}\) with operators \(T_i \in S\). We note \(\mathcal{H}(\mathcal{X})\) the set of non-empty compacts of \(\mathcal{X}\). \(\mathcal{H}(\mathcal{X})\) is a complete metric space with the Hausdorff distance. The associated Hutchinson operator is:

\[
K \in \mathcal{H}(\mathcal{X}) \mapsto TK = T_0K \cup ... \cup T_{N-1}K.
\]

This operator is contractive in the complete metric space \(\mathcal{H}(\mathcal{X})\) and admits a fixed point, called attractor [2]:

\[
\mathcal{A}(T) = \lim_{n \to \infty} T^nK \text{ with } K \in \mathcal{H}(\mathcal{X}).
\]

By introducing a finite set \(\Sigma\), the IFS can be indexed \(T = (T_i)_{i \in \Sigma}\) and the attractor \(\mathcal{A}(T)\) has an address function [2, 4] defined on \(\Sigma^\omega\), the set of infinite words of \(\Sigma\):

\[
\rho \in \Sigma^\omega \mapsto \phi(\rho) = \lim_{n \to \infty} T_{\rho_1}...T_{\rho_n} \lambda \in \mathcal{X} \text{ with } \lambda \in \mathcal{X}.
\]  

(1)

2.2 Approximation formulation

**Corollary 1.** Every address function associated with an IFS is in \(L^2(\Sigma^\omega, \mathcal{X})\).

**Proof.** See [7].

In the Hilbert space of address functions, optimization problem can be expressed with the following formulation. Let \(\phi\) be an address function, find \(T\) that minimizes the error function:

\[
T \in S^\Sigma \mapsto g(T) = ||\Psi(T) - \phi||_2^2 \in \mathbb{R}_+
\]

with \(\Psi(T)\) the address function associated with \(T\).

To apply standard non-linear fitting methods, the function \(g\) needs to have good properties. This function is a quadratic form of \(\Psi\). In the following, we will expose these properties.
2.3 Affine IFS

We now deal with affine IFS, that means IFS defined with affine contractions in $\mathcal{X} = \mathbb{R}^m$. In this case, the contractive semigroup can be characterized. An affine operator is defined by a couple $(u, L)$ with $u \in \mathbb{R}^m$ and $L$ a $m \times m$ matrix:

$$ Tp = u + Lp $$

The set of affine operators acting on $\mathbb{R}^m$ is a complete metric space with the following distance:

$$ d(T, T') = \|u - u'\| + \|L - L'\| $$

where

$$ \|L\| = \max_{\|u\|=1} \|Lu\|.$$

**Proposition 1.** The affine contractive semigroup is an open set $S = \mathbb{R}^m \times B_1$ where $B_1 = \{L/\|L\| < 1\}$.

**Proof.** One can easily verify that $T \in S$ implies its contraction:

$$ \exists r \in [0, 1[, \forall p, q \in \mathbb{R}^m, d(Tp, Tq) \leq rd(p, q) $$

$$ \Leftrightarrow \exists r \in [0, 1[, \forall u \in \mathbb{R}^m, \|Lu\| \leq r\|u\| $$

$$ \Leftrightarrow \exists r \in [0, 1[, \forall u \in \mathbb{R}^m, \|u\| = 1, \|Lu\| \leq r $$

$$ \Leftrightarrow \max_{\|u\|=1} \|Lu\| < 1 $$

2.4 Analyticity

In this section, we precise property of the function:

$$ \psi : \mathcal{S}^{\Sigma} \rightarrow \psi(\mathcal{T}) \in C^0(\Sigma^\omega, \mathcal{X}) $$

**Definition 1.** Let $\mathcal{S}^{\Sigma}$ be the set of indexed IFS $\mathcal{T} = (T_i)_{i \in \Sigma}$. Let $\psi_p$ be the following function, where $p \in \Sigma^\omega$ is fixed:

$$ \psi_p : \mathcal{S}^{\Sigma} \rightarrow \mathcal{X} $$

$$ \mathcal{T} \mapsto \psi_p(\mathcal{T}) = \lim_{n \rightarrow \infty} T_{\rho_1} \cdots T_{\rho_n}p $$

As $T_i$ is affine, we may decompose it in a translation vector $u_i$ and a linear part $L_i$:

$$ T_ip = u_i + L_ip $$

In this case, the product $T_iT_j$ gives $u_i + L_iu_j$ as translation vector and $L_iL_j$ as linear part. Then, we expand the matrix product:

$$ T_{\rho_1}\dots T_{\rho_n}p = u_{\rho_1} + L_{\rho_1}u_{\rho_2} + L_{\rho_1}L_{\rho_2}u_{\rho_3} + \cdots + L_{\rho_1}\cdots L_{\rho_{n-1}}u_{\rho_n} + L_{\rho_1}\cdots L_{\rho_n}p $$

(2)
When \( n \) tends to infinity, \( p \) has no influence on this formula:

\[
\lim_{n \to \infty} (L_{\rho_1} \ldots L_{\rho_n} p) = 0
\]

because \( L_i \) are linear contractions. Then \( \psi_{\rho}(\mathcal{T}) \) can be written as a summation:

\[
\psi_{\rho}(\mathcal{T}) = \lim_{n \to \infty} \sum_{k=1}^{n} L_{\rho_1} \ldots L_{\rho_{k-1}} u_{\rho_k}
\]

**Proposition 2.** For every \( \rho \) in \( \Sigma^\omega \), the function \( \psi_{\rho} \) is analytical on \( S^\Sigma \).

*Proof.* See [7].

**Proposition 3.** The function:

\[
\psi : S^\Sigma \to L^2(\Sigma^\omega, \mathcal{X})
\]

is analytical.

*Proof.* The function \( \psi \) is a family of functions:

\[
\psi(\mathcal{T}) = (\psi_{\rho}(\mathcal{T}))_{\rho \in \Sigma^\omega}.
\]

The proof of analyticity of \( \psi_{\rho} \) based on differentials is valid with \( \psi \) when introducing \( \psi \) as a function of both \( \mathcal{T} \) and \( \rho \):

\[
d^k \psi(\mathcal{T})(\rho) = d^k \psi_{\rho}(\mathcal{T}).
\]

### 2.5 Error Estimation

In practical, this error function is approximated on samples, that means on a finite number of values:

\[
g(\mathcal{T}) \approx g_n(\mathcal{T}) = \frac{1}{N^n} \sum_{\alpha \in \Sigma^n} \frac{1}{M} \sum_{k \in \Omega} \| \psi_{\alpha k^\omega}(\mathcal{T}) - \varphi(\alpha k^\omega) \|^2
\]

We toggle from a functional distance to a tabulation distance.

To perform finite exact computations, we take advantage of the fact that each transformation has a fixed point:

\[
T_k c_k = c_k
\]

We evaluate the function at a deep \( n \), with \( |\alpha| = n \). Then, the function has the form:

\[
\psi_{\alpha k^\omega}(\mathcal{T}) = T_{\alpha_1} \ldots T_{\alpha_n} \phi(k^\omega),
\]

\[
= T_{\alpha_1} \ldots T_{\alpha_n} c_k.
\]

In this case, only polynomial computations have to be performed, \( g_n \) is a polynomial function:

\[
g_n(\mathcal{T}) = \frac{1}{N^n} \sum_{\alpha \in \Sigma^n} \frac{1}{M} \sum_{k \in \Omega} \| T_{\alpha_1} \ldots T_{\alpha_n} c_k - \varphi(\alpha k^\omega) \|^2
\]
2.6 Resolution

We proved the analyticity of affine IFS functions with respect to their matrix coefficients. We can now use a differential method to solve our problem. The literal derivative is more complex to evaluate than a numerical approximation with a perturbation. The optimization algorithm used is LEVENBERG-MARQUARDT, an improved gradient method [15].

3 Function Approximation

This section will show a very simple example of numerical optimization using affine IFS defined in $\mathbb{R}$.

3.1 Model overview

Let $\Sigma = \{0, \ldots, N - 1\}$. Transformations operate on $\mathbb{R}$:

$$T_i : \mathbb{R} \to \mathbb{R}$$

$$x \mapsto a_i x + b_i$$

Each transformation is defined by two scalars. In this case, the address function is:

$$\phi(\rho) = b_{\rho_1} + a_{\rho_1} b_{\rho_2} + a_{\rho_1} a_{\rho_2} b_{\rho_3} + \cdots$$

A simple series converge to this value:

$$\phi(\rho) = \lim_{n \to \infty} B_n$$

where

$$\begin{cases} B_1 = b_{\rho_1} \\ B_{i+1} = B_i + A_i b_{\rho_{i+1}} \text{ for } i \geq 1 \end{cases}$$

and

$$\begin{cases} A_1 = a_{\rho_1} \\ A_{i+1} = A_i a_{\rho_{i+1}} \text{ for } i \geq 1 \end{cases}$$

Remark 1. This kind of IFS is not Fractal Interpolation Functions since they are defined in $\mathbb{R}$ (FIF are defined in $\mathbb{R}^2$).

3.2 Approximation formulation

When dealing with approximation, a common data type is an ordered list of points $(x_i, y_i)_{i=1,\ldots,p}$. The value of $x_i$ will be used to extract an address associated to the sample, whereas the value of $y_i$ will be the target value of the address function. Let $a^{(i)} = a_{1}^{(i)} \ldots a_{n}^{(i)}$ be the $N$-adic expansion of $\bar{x}_i$ with $x_i = \bar{x}_i + \epsilon_i$ and $\epsilon_i < \frac{1}{N^{n+1}}$: 
\[ \bar{x}_i = \sum_{j=1}^{n} \frac{1}{N_j} \alpha^{(i)}_j \]

Then, the approximation problem with affine IFS in \( \mathbb{R} \) can be formulated.

Given data entries \((x_i, y_i)_{i=1,...,p}\) where \(x_i+1 > x_i\), and a number of transformations \(N\), find the IFS that minimizes the error:

\[
T_{opt} = \arg\min_{T \in \mathcal{S}} g_n(T) = \arg\min_{T \in \mathcal{S}} \frac{1}{p} \sum_{i=1}^{p} \left( \psi_{\alpha_1^{(i)} ... \alpha_n^{(i)}}(T) - y_i \right)^2
\]

### 3.3 Results

We have tested our approximation method on several data sets, ranging from smooth curves to random data. As expected, the approximation quality depends on the number of transformations \(N\) taken.

Figure 1 shows the approximation of a cubic curve \(y = 6(x - \frac{1}{2})^3\) with the method described previously. In these graphs, x-coordinates represents the address values and y-coordinates the values of the address function. The original curve contains 1000 points. When approximating with only 2 transformations, the fitting is not good. When the number of transformations becomes larger, the quality of approximation is better.

Figure 2 shows the approximation of a random function that contains 100 points. With only 5 transformations, the result is not so bad. Increasing the number of transformations leads to a better approximation. The upper limit of \(N\) is when we reach the number of data points: \(N = p\). In this case, the exact reconstruction is possible. The method used to solve the approximation problem is not global. It means that the result can be a local minimum.

### 4 Modelisation of rough shapes

In order to propose an efficient solution to the problem of rough surface approximation, we have used a parametric model based on a fractal model. In [18, 20], we have proposed a projected IFS model for fractal curve and surfaces. This model combines a fractal classical approach – Iterative Function Systems – and CAGD classical approach – free form based on control points. These points allow an easy and flexible control of the fractal shape generated by the IFS model and provide a high quality fitting.

#### 4.1 Projected IFS model

To allow more flexible modeling, we introduced and used a projected IFS model [18, 19]. The way to obtain projected IFS attractors is to use a barycentric metric space \( \mathcal{X} = B^J \):
Then, the iteration semigroup is constituted of matrices with barycentric columns:

\[ S_J = \{ T \mid \sum_{j \in J} T_{ij} = 1, \forall i \in J \} \]

This choice leads to the generalization of IFS attractors named \textit{projected IFS attractors}:

\[ PA(T) = \{ P\lambda \mid \lambda \in A(T) \} \]

where \( P \) is a polygon or grid of control points \( P = (p_j)_{j \in J} \) and \( P\lambda = \sum_{j \in J} \lambda_j p_j \).

The associated address function is:
\[ \varphi(\rho) = P\varphi(\rho) = \sum_{i \in J} p_i \phi_i(\rho) \]

As shown in figure 3, a two-dimensional addressing can be easily calculated by a P\'{E}ANO code mapping.

![P\'{E}ANO code](image)

**Fig. 3. P\'{E}ANO code with \( \Sigma = \{0, 1, 2, 3\} \)**

In figure 4, an example of bivariate function generated by projected IFS is shown. This function defines a surface, projected through a 4 \( \times \) 4 control points grid. Here, control points are scalars \( p_i = z_i \in \mathbb{R} \):

\[ \varphi : \Sigma^\omega \to \mathbb{R} \]
\[ \rho \mapsto \varphi(\rho) = \sum_{i \in \omega} z_i \phi_i(\rho) \]

The construction of the projected attractor is determinist: it only requires recursive subdivisions as shown in figure 4.

4.2 Projected IFS tree model

Natural objects are composed of heterogeneous parts. To cope with this problem, we introduced another generalization: projected IFS trees model [12, 6]. Let \( \Gamma \) be a cut of the tree \((\Sigma^\infty, \leq)\), that means a finite part of \( \Sigma^* \) such that each word \( \rho \in \Sigma^\omega \) admits a unique decomposition on \( \Gamma \times \Sigma^\omega \):

\[ \rho = \gamma \tau \text{ with } \gamma \in \Gamma \text{ and } \tau \in \Sigma^\omega. \]

If we denote \( m = \max_{\gamma \in \Gamma} |\gamma| \), then we have the following decomposition:

\[ \forall n \geq m \quad \Sigma^n = \bigcup_{\gamma \in \Gamma} \gamma \Sigma^{n-|\gamma|} \quad \text{and} \quad \Sigma^\omega = \bigcup_{\gamma \in \Gamma} \gamma \Sigma^\omega \]

Drawn from the families:

- of address functions \( \phi^\gamma \in C^0(\Sigma^\omega, \mathcal{X}_\gamma) \),
- of affine functions \( P^\gamma : \mathcal{X}_\gamma \to \mathcal{X} \),

we use the following address function to modelize surfaces:

\[ \phi(\gamma \tau) = P^\gamma \phi^\gamma(\tau) \]

and:

\[ \forall \gamma \in \Gamma, \forall i \in \Sigma, \phi^\gamma(i \tau) = T_i^\gamma \phi^\gamma(\tau). \]
Proposition 4. Every address function $\phi^\gamma$ built on address functions associated with IFS $T^\gamma$ is in $L^2(\Sigma^\omega, \mathcal{X})$ and $\phi \in L^2(\Sigma^\omega, \mathcal{X})$.

Proof. The functions $\phi^\gamma$ are associated with IFS, that means that they verify the decreasing condition, and $\phi$ too (see [7]).

An example of heterogeneous surface is given in figure 5. Each patch of the surface can have different properties. In this example, we have mixed rough and smooth modeling together.
4.3 Approximation formulation

We want to approximate data entries arranged in grids \((z_{i,j})_{i,j\in[0\ldots2^n]}\) with a projected IFS tree \((P, T) = (P^\gamma, T^\gamma)_{\gamma\in\Gamma}\). Given a tree cut \(\Gamma\), the model is described by two families of parameters: \((P^\gamma)_{\gamma\in\Gamma}\) and \((T^\gamma)_{\gamma\in\Gamma}\). The address is split into two parts: the leaf \(\gamma \in \Gamma\), address of the projected IFS model, and \(\tau \in \Sigma^\omega\) address of the point in the projected IFS model:

\[
\psi^\Gamma_{\gamma\tau}(P, T) = P^\gamma \psi(\tau)
\]

\(\psi^\Gamma\) is analytical with respect to \(T = (T^\gamma)_{\gamma\in\Gamma}\) and affine with respect to \(P = (P^\gamma)_{\gamma\in\Gamma}\).

The approximation algorithm has to perform simultaneously two tasks: find the tree cut \(\Gamma\) and the associated projected IFS models \((P^\gamma, T^\gamma)\). To satisfy this constraint, we have constructed another norm combining a maximum through the tree cut with a quadratic norm:

\[
||\psi^\Gamma(P, T) - \varphi|| = \max_{\gamma\in\Gamma} ||P^\gamma \psi(\tau) - \varphi^\gamma||
\]

with \(\varphi^\gamma(\tau) = \varphi(\gamma\tau)\).

The algorithm is then implemented using a threshold \(\epsilon\) that indicates the maximum value allowed in a leaf. If this constraint is not satisfied, the leaf is split into four, recursively (see figure 6). Then, the whole norm is smaller or equal to the threshold:

\[
||\psi^\Gamma(P, T) - \varphi|| \leq \epsilon \iff \forall \gamma \in \Gamma, ||P^\gamma \psi(\tau) - \varphi^\gamma|| \leq \epsilon
\]
The algorithm is recursive and uses only locally the analyticity property of the error function. The goal is to find the minimal cut that satisfies the constraint:

$$\Gamma_{\epsilon} = \min \left\{ \Gamma / \| \psi^R(P, T) - \varphi \| \leq \epsilon \right\}$$

We construct an address function tabulation associated to the data entries:

$$\varphi(\alpha k^w) = \varphi((\alpha' k^w') \bullet (\alpha'' k^w''))$$

$$= z_{j',j''},$$

with

$$j' = \sum_{l=1}^{n} 2^{n-l} \alpha'_l$$

$$j'' = \sum_{l=1}^{n} 2^{n-l} \alpha''_l$$

and $$\alpha', \alpha''$$ verifies $$\alpha' \bullet \alpha'' = \alpha'_1 \bullet \alpha''_1 \ldots \alpha'_n \bullet \alpha''_n,$$ with $$\alpha'_1 \bullet \alpha''_1 = 2\alpha'_1 + \alpha''_1$$ and $$k = k' \bullet k''$$.

Error estimation for a given leaf $$\gamma \in \Gamma$$ is:

$$g_n(P^\gamma, T^\gamma) = \frac{1}{4^n - \gamma} \sum_{\alpha \in \Sigma^{n-\gamma}} \frac{1}{M} \sum_{k \in \Omega} (P^\gamma \psi_{\alpha k^w} (T^\gamma) - \varphi(\gamma \alpha k^w))^2$$

### 4.4 Surface reconstruction

Figure 7 represents the result of a surface approximation. The data is an elevation grid of size 257 × 257 extracted from Digital Terrain Elevation Data (DTED) Level 0.\(^3\) In this example, the approximation method has been applied with an error threshold based on a minimum local PSNR value. PSNR

\(^3\)Data available at http://data.geocomm.com/catalog/FR/group121.html
is directly related to the definition of $g_n(P^\gamma, T^\gamma)$:

$$\text{PSNR}(P^\gamma, T^\gamma) = 10 \log_{10}(\frac{\max g_n(P^\gamma, T^\gamma)}{g_n(P^\gamma, T^\gamma)})$$

where max is the range of input data.

![Fig. 7. Approximation of the French “Massif central” mountain](image)

(a) Original surface  
(b) Approximated surface

In this example, the threshold $\epsilon$ is such that for all $\gamma$ in $\Gamma$ the value of PSNR($P^\gamma, T^\gamma$) is greater than 40dB.

### 4.5 Image Compression

By using the same model, we are able to perform image compression. The input data is a greyscale grid of size $257 \times 257$. The difference is in the approximation method, that optimizes the rate/distortion ratio. Figure 8 shows an example of image compression. For a bit rate of 0.12bpp, the corresponding error is PSNR=28.3dB, with the following classical definition of PSNR:

$$\text{PSNR}(P, T) = 10 \log_{10}(\frac{255}{g_n(P, T)})$$

where $g_n(P, T) = \sum_{\gamma \in \Gamma} g_n(P^\gamma, T^\gamma)$ corresponds to the estimation of quadratic distance:

$$g_n(P, T) \approx \|\psi^\Gamma(P, T) - \varphi\|_2^2$$

Detailed method is available in [12, 6].
5 Conclusion

We showed that analytical approach and methods using derivation properties can be used to perform the fractal inverse problem. This problem can be formulated as an optimization problem in an Hilbert space. For a useful family of fractal model based on affine IFS, the error function is analytical. Hence, the optimization problem has a non-linear classical formulation. Methods based on non-linear optimization algorithms can be applied with interesting numerical results in surface reconstruction and image compression.

References


