Simulate, Animate, Illuminate Cloud Scenes

Albert Benassi   Pascal Bleuyard   Frédéric Szczap
Yaya Gour

A.Benassi@opgc.univ-bpclermont.fr

Laboratoire de Météorologie Physique (LaMP)
Université Blaise Pascal, CNRS, Aubière, France

I3RC, Kiel, Germany, October 11–14, 2005

Outline

Outline

Simulate Cloud Fields?

Why cloud fields simulation?

In Science: Global Circulation Models (GCM), Meteorology, Instrumentation…

In Image Processing: Texture Generation, Animation…

Simulate Cloud Fields?

Why cloud fields simulation?

In Science: Global Circulation Models (GCM), Meteorology, Instrumentation. . .

In Image Processing: Texture Generation, Animation. . .

Scientifi c Simulation

Science

- In GCM, the pixel size is 100 km $\times$ 100 km at least.
- To compute the energy budget, we must take into account the radiative transfer.
  We must assigne to the pixel an “effective transfer” value.
- We need a subpixel model.

Scientifi c Simulation

Science

- In GCM, the pixel size is 100 km $\times$ 100 km at least.
- To compute the energy budget, we must take into account the radiative transfer.
  We must assigne to the pixel an “effective transfer” value.
- We need a subpixel model.

Scientifi c Animation

Our purpose is to generate “low-cost” realistic cloud fields, both static, dynamic and illuminated.
The model must be Multi-Scale.

Scientifi c Animation

Our purpose is to generate “low-cost” realistic cloud fields, both static, dynamic and illuminated.
The model must be Multi-Scale.

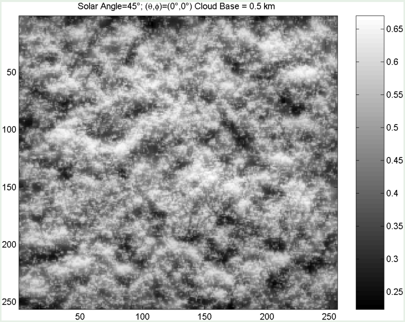
Scientifi c Animation

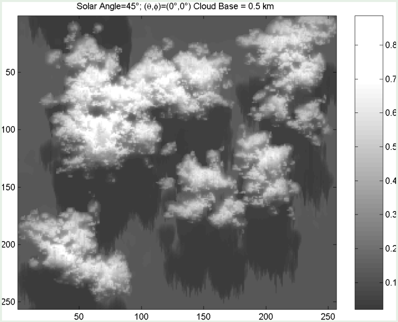
Our purpose is to generate “low-cost” realistic cloud fields, both static, dynamic and illuminated.
The model must be Multi-Scale.

Scientifi c Animation

Our purpose is to generate “low-cost” realistic cloud fields, both static, dynamic and illuminated.
The model must be Multi-Scale.

Illuminated Cloud Fields





Outline

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$\Omega.N(r,\Omega) = -\sigma(r)[N(r,\Omega) - J(r,\Omega)]$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega^{'})$ the phase function at r scattering in the direction Ω from direction $\Omega^{'}$

+ Boundary Conditions

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$\Omega.N(r,\Omega) = -\sigma(r)[N(r,\Omega) - J(r,\Omega)]$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega^{'})$ the phase function at r scattering in the direction Ω from direction $\Omega^{'}$

+ Boundary Conditions

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$\Omega.N(r,\Omega) = -\sigma(r)[N(r,\Omega) - J(r,\Omega)]$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega^{'})$ the phase function at r scattering in the direction Ω from direction $\Omega^{'}$

+ Boundary Conditions

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$\Omega.N(r,\Omega) = -\sigma(r)[N(r,\Omega) - J(r,\Omega)]$$

$$J(r,\Omega) = \frac{\varpi(r)\sigma(r)}{4\pi}\int P(r,\Omega,\Omega^{'})N(r,\Omega^{'})d\omega(\Omega^{'}) + [1-\varpi]B(r)$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega^{'})$ the phase function at r scattering in the direction Ω from direction $\Omega^{'}$

+ Boundary Conditions

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$J(r,\Omega) = \frac{\varpi(r)\sigma(r)}{4\pi} \int P(r,\Omega,\Omega^{'})N(r,\Omega^{'})d\omega(\Omega^{'}) + [1-\varpi]B(r)$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega^{'})$ the phase function at r scattering in the direction Ω from direction $\Omega^{'}$

+ Boundary Conditions

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$J(r,\Omega) = \frac{\varpi(r)\sigma(r)}{4\pi} \int P(r,\Omega,\Omega')N(r,\Omega')d\omega(\Omega') + [1-\varpi]B(r)$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega')$ the phase function at r scattering in the direction Ω from direction Ω'

+ Boundary Conditions

Illuminating Cloud Fields

Solving the RTE for illuminating cloud fields:

Mathematical Formulation

$$J(r,\Omega) = \frac{\varpi(r)\sigma(r)}{4\pi} \int P(r,\Omega,\Omega^{'})N(r,\Omega^{'})d\omega(\Omega^{'}) + [1-\varpi]B(r)$$

- N the radiance at r in direction Ω
- Ω directional unit at point r
- J the source function at r in direction Ω
- σ the extinction coefficient at point r
- ϖ the single scatering albedo
- B the Plank function at r
- $P(r,\Omega,\Omega^{'})$ the phase function at r scattering in the direction Ω from direction $\Omega^{'}$

+ Boundary Conditions

Solving the RTE ?

- It is a non linear problem, because the cloud is part of the data and because of strong inter-scale interactions.
- So it is a challenging problem, specially in 3D inhogeneous case.
- Analysing the scale interactions may help finding a new way for illumination.
- Multi-resolution analysis is used for solving RTE (Ferlay et al 2005) and to look at those interactions.
- It is very costly to solve RTE!

Mutiresolution Analysis

Mathematical Formulation

Let $\varphi_k, \psi^{\varepsilon}_{j,k}; j \in N^*,\ k \in \mathbb{Z}^3,\ \varepsilon \in \{0, ., 7\}$ a MR

- φ is the scale function
- ψ^{ε} are the mother wavelets
- μ, ν connexion's indices

Analysis of Energy Transfers through Tensor T
Non linearties similar to those of NS Equation
Functions of variable Ω are analysed with Spherical Harmonics (Evans 1998)
Solving RTE using MR is made in Ferlay et al (2005)

Mutiresolution Analysis

Mathematical Formulation

Let $\varphi_k, \psi^{\varepsilon}_{j,k}; j \in N^*,\ k \in \mathbb{Z}^3,\ \varepsilon \in \{0,.,7\}$ a MR

- φ is the scale function
- ψ^{ε} are the mother wavelets
- μ, ν connexion's indices

Analysis of Energy Transfers through Tensor T
Non linearties similar to those of NS Equation
Functions of variable Ω are analysed with Spherical Harmonics (Evans 1998)
Solving RTE using MR is made in Ferlay et al (2005)

Mutiresolution Analysis

Mathematical Formulation

- φ is the scale function
- ψ^{ε} are the mother wavelets
- μ, ν connexion's indices

$$N(r,\Omega) = \sum_k \varphi_k(r) n_k(\Omega) + \sum_{j,k,\varepsilon} \psi^{\varepsilon}_{j,k}(r) n^{\varepsilon}_{j,k}(\Omega)$$
$$\alpha(r,\Omega) = \sum_k \varphi_k(r) \alpha_k(\Omega) + \sum_{j,k,\varepsilon} \psi^{\varepsilon}_{j,k}(r) \alpha^{\varepsilon}_{j,k}(\Omega)$$
$$(N.\alpha)^{\varepsilon}_{j,k} = \sum_{\mu,\nu} T^{\lambda}_{\mu,\nu} n_{\mu} \alpha_{\nu}, \ \lambda = (\varepsilon, j, k)$$

Analysis of Energy Transfers through Tensor T

Non linearties similar to those of NS Equation

Functions of variable Ω are analysed with Spherical Harmonics (Evans 1998)

Solving RTE using MR is made in Ferlay et al (2005)

Fondamental Remark

The set

$$(j \in N^*, k \in \mathbb{Z}^3)$$

can be seen geometrically as a

TREE

Outline

tdMAP: a Density Field Generator

From the Multi-Resolution Analysis of Fractional Brownian Motion (Benassi 1995, Benassi et al 1997)

Mathematical Formulation

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_{j,k}} F_{j,k}(2^j x - k)\xi_{j,k} \text{ with:}$$

- X the generated density field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $H_\square$ the Hurst exponent;
- $F_\square$ the "morphlet" function;
- $\xi_\square$ a family of random values.

tdMAP: a Density Field Generator

From the Multi-Resolution Analysis of Fractional Brownian Motion (Benassi 1995, Benassi et al 1997)

Mathematical Formulation

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_{j,k}} F_{j,k}(2^j x - k)\xi_{j,k} \text{ with:}$$

- X the generated density field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $H_\square$ the Hurst exponent;
- $F_\square$ the "morphlet" function;
- $\xi_\square$ a family of random values.

tdMAP: a Density Field Generator

From the Multi-Resolution Analysis of Fractional Brownian Motion (Benassi 1995, Benassi et al 1997)

Mathematical Formulation

$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_{j,k}} F_{j,k}(2^j x - k)\xi_{j,k}$ with:

- X the generated density field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $H_\square$ the Hurst exponent;
- $F_\square$ the "morphlet" function;
- $\xi_\square$ a family of random values.

tdMAP: a Density Field Generator

From the Multi-Resolution Analysis of Fractional Brownian Motion (Benassi 1995, Benassi et al 1997)

Mathematical Formulation

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square \text{ with:}$$

- X the generated density field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $H_\square$ the Hurst exponent;
- $F_\square$ the "morphlet" function;
- $\xi_\square$ a family of random values.

tdMAP: a Density Field Generator

From the Multi-Resolution Analysis of Fractional Brownian Motion (Benassi 1995, Benassi et al 1997)

Mathematical Formulation

$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$ with:

- X the generated density field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $H_\square$ the Hurst exponent;
- $F_\square$ the "morphlet" function;
- $\xi_\square$ a family of random values.

tdMAP: a Density Field Generator

From the Multi-Resolution Analysis of Fractional Brownian Motion (Benassi 1995, Benassi et al 1997)

Mathematical Formulation

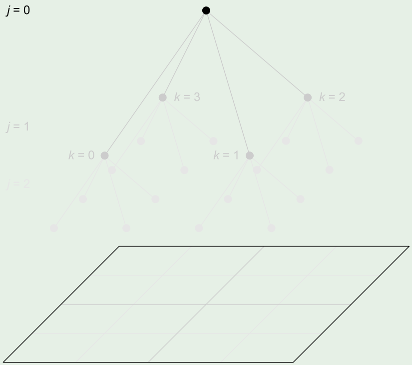
$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square \text{ with:}$$

- X the generated density field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $H_\square$ the Hurst exponent;
- $F_\square$ the "morphlet" function;
- $\xi_\square$ a family of random values.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

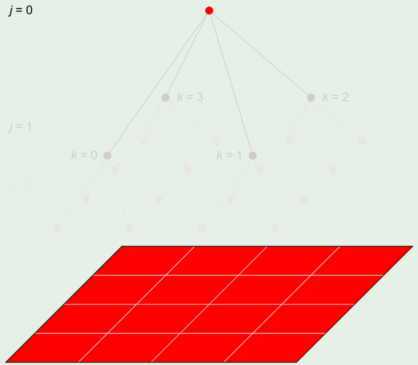


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

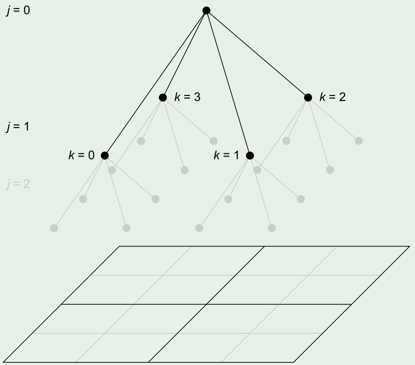


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

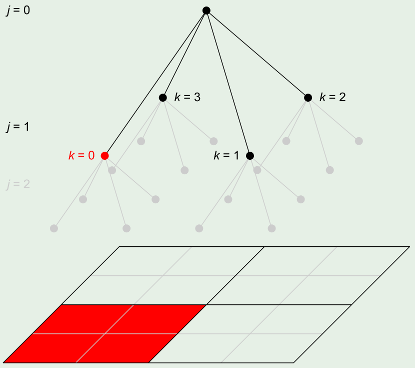


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

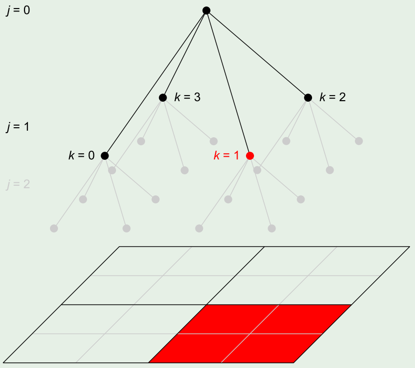


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

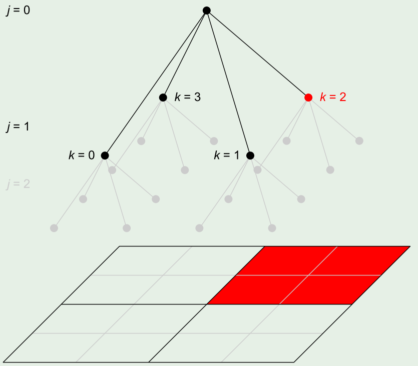


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

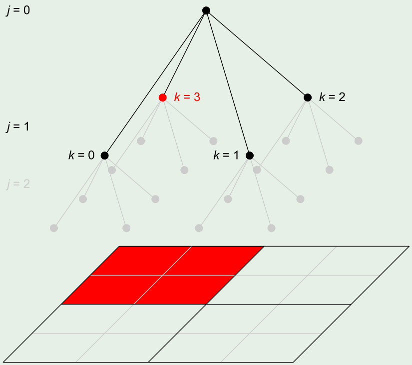


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

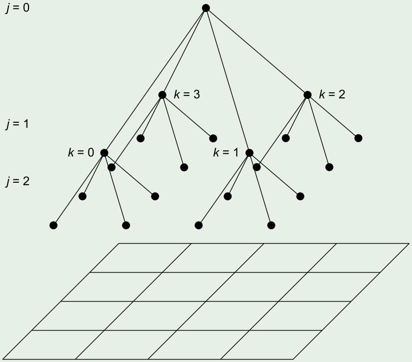


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

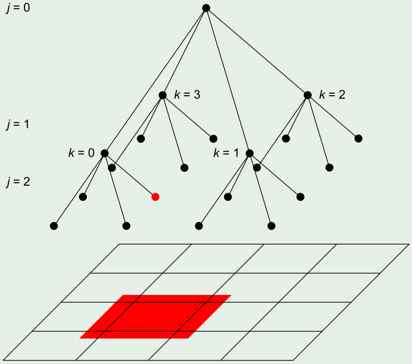


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

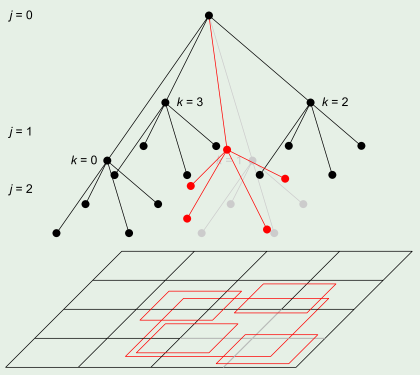


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

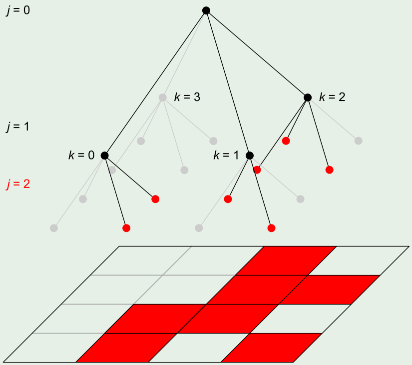


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Coding Tree

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_\square} F_\square(2^j x - k)\xi_\square$$

2 dimensional domain

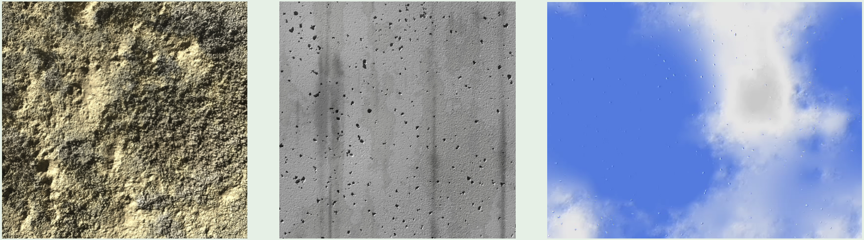


- The backbone of tdMAP is a decorated tree;
- At each node of the tree are attached some objects ($H_\square$, $F_\square$, and $\xi_\square$) and actions (swelling, shifting, pruning) that can be freely set at any scale and spatial position.

Confi gurating tdMAP

In order to configure tdMAP, one can use both observations and measures.

Photorealistic Textures from Allegorithmic

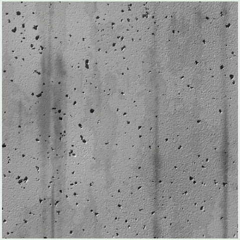
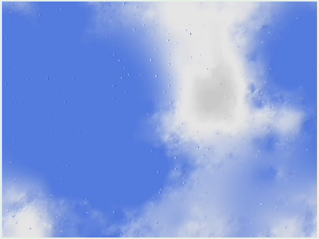


web address: `http://www.allegorithmic.com`

Confi gurating tdMAP

In order to configure tdMAP, one can use both observations and measures.

Photorealistic Textures from Allegorithmic

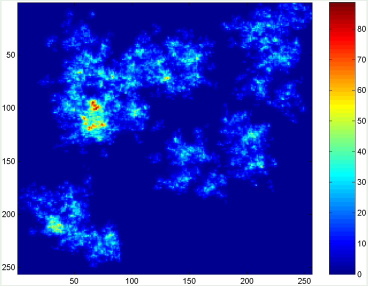




web address: `http://www.allegorithmic.com`

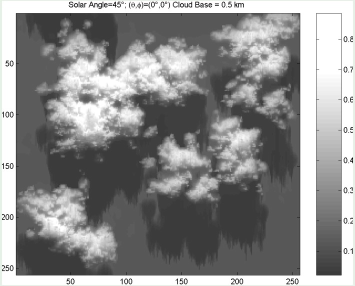
Low-level Clouds Generation

With tdMAP it is possible to build 2D stratocumulus and cumulus cloud fields with reasonably realistic statistical properties (BENASSI et al., 2004).

Fractional Coverage Cloud



Optical depth

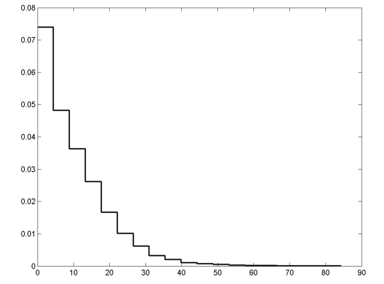


Top radiances

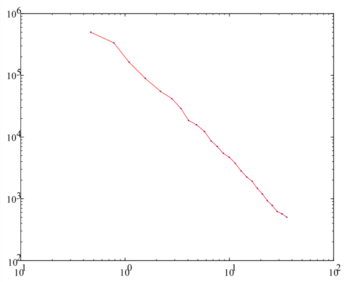
Low-level Clouds Generation

With tdMAP it is possible to build 2D stratocumulus and cumulus cloud fields with reasonably realistic statistical properties (BENASSI et al., 2004).

Fractional Coverage Cloud



Optical depth PDF ($\rho_\tau \approx 1.6$)



Fourier power spectrum ($\beta \approx -1.78$)

Morphlet Transform

Mathematical Formulation

$$F(x) = \sum_{\varepsilon,j,k} f^{\varepsilon}_{j,k} \Delta^{\varepsilon}_{j,k}(x)$$

- F is the signal (image)
- in d dimensions $\varepsilon \in \{0, 1\}^d - \{1, .., 1\}$
- $f^{\varepsilon}_{j,k}$ are the morphlets coefficients
- $\Delta(x)$ is a pyramid function supported in $[0, 1]^d$
- $\Delta^{\varepsilon}_{j,k}(x) = \Delta(2^j x - k - \varepsilon/2)$

Morphlet coefficients are obtained hierarchically from values of F on $2^{-J}\mathbb{Z}^d$.
This decomposition gives a hierarchy of “facets”.

Morphlet Transform

Mathematical Formulation

$$F(x) = \sum_{\varepsilon,j,k} f^{\varepsilon}_{j,k} \Delta^{\varepsilon}_{j,k}(x)$$

- F is the signal (image)
- in d dimensions $\varepsilon \in \{0,1\}^d - \{1,..,1\}$
- $f^{\varepsilon}_{j,k}$ are the morphlets coefficients
- $\Delta(x)$ is a pyramid function supported in $[0,1]^d$
- $\Delta^{\varepsilon}_{j,k}(x) = \Delta(2^j x - k - \varepsilon/2)$

Morphlet coefficients are obtained hierarchically from values of F on $2^{-J}\mathbb{Z}^d$.
This decomposition gives a hierarchy of “facets”.

Morphlet Transform

Mathematical Formulation

$$F(x) = \sum_{\varepsilon,j,k} f^{\varepsilon}_{j,k} \Delta^{\varepsilon}_{j,k}(x)$$

- F is the signal (image)
- in d dimensions $\varepsilon \in \{0,1\}^d - \{1,..,1\}$
- $f^{\varepsilon}_{j,k}$ are the morphlets coefficients
- $\Delta(x)$ is a pyramid function supported in $[0,1]^d$
- $\Delta^{\varepsilon}_{j,k}(x) = \Delta(2^j x - k - \varepsilon/2)$

Morphlet coefficients are obtained hierarchically from values of F on $2^{-J}\mathbb{Z}^d$.

This decomposition gives a hierarchy of “facets”.

Morphlet Transform

Mathematical Formulation

$$F(x) = \sum_{\varepsilon,j,k} f^{\varepsilon}_{j,k} \Delta^{\varepsilon}_{j,k}(x)$$

- F is the signal (image)
- in d dimensions $\varepsilon \in \{0,1\}^d - \{1,..,1\}$
- $f^{\varepsilon}_{j,k}$ are the morphlets coefficients
- $\Delta(x)$ is a pyramid function supported in $[0,1]^d$
- $\Delta^{\varepsilon}_{j,k}(x) = \Delta(2^j x - k - \varepsilon/2)$

Morphlet coefficients are obtained hierarchically from values of F on $2^{-J}\mathbb{Z}^d$.

This decomposition gives a hierarchy of “facets”.

Morphlet Transform

Mathematical Formulation

$$F(x) = \sum_{\varepsilon,j,k} f^{\varepsilon}_{j,k} \Delta^{\varepsilon}_{j,k}(x)$$

- F is the signal (image)
- in d dimensions $\varepsilon \in \{0,1\}^d - \{1,..,1\}$
- $f^{\varepsilon}_{j,k}$ are the morphlets coefficients
- $\Delta(x)$ is a pyramid function supported in $[0,1]^d$
- $\Delta^{\varepsilon}_{j,k}(x) = \Delta(2^j x - k - \varepsilon/2)$

Morphlet coefficients are obtained hierarchically from values of F on $2^{-J}\mathbb{Z}^d$.

This decomposition gives a hierarchy of “facets”.

Mathematical Formulation

Theorem

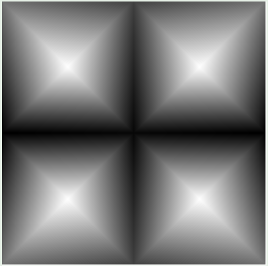
For $\beta, 0 < \beta < 1$, the morphlet decomposition of any $F \in H^{\beta}(\mathbb{R}^d)$ is unique.
On every compact, the associated serie of morphlets converges uniformly to F.

Morphlet Decomposition: Examples

Lena

Original

Details



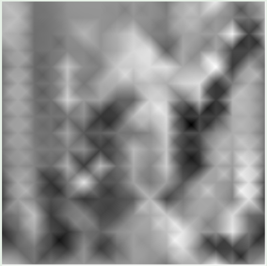
Approximation

Morphlet Decomposition: Examples

Lena

Original

Details

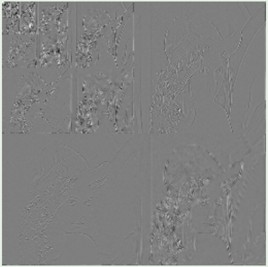


Approximation

Morphlet Decomposition: Examples

Lena

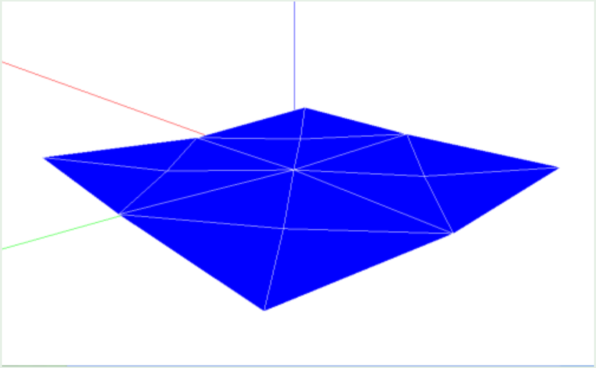

Original


Details


Approximation

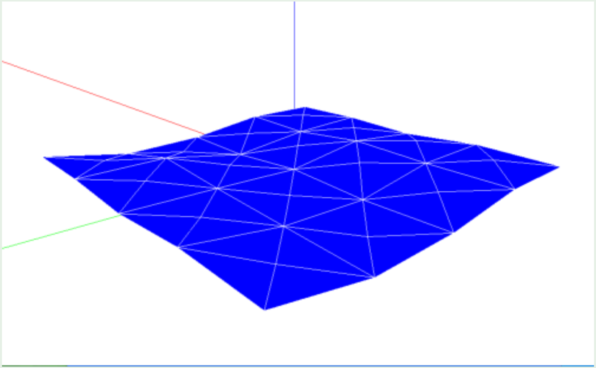
Morphlet Decomposition: Examples

Sea Surface Facetization



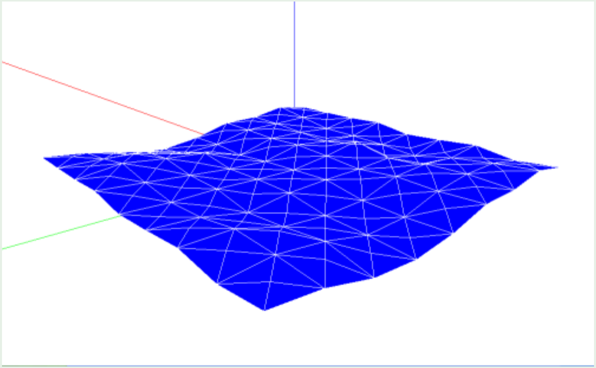
Morphlet Decomposition: Examples

Sea Surface Facetization



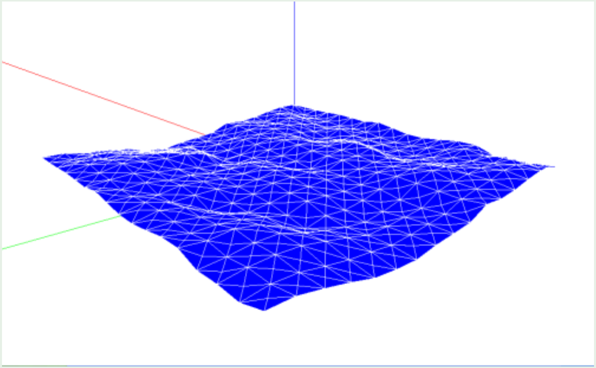
Morphlet Decomposition: Examples

Sea Surface Facetization



Morphlet Decomposition: Examples

Sea Surface Facetization



Outline

vtdMAP: a Vector Field Generator

The study of Operator Self-Similar Gaussian Processes with Stationary Increments (BAHADORAN, BENASSI and DEBICKI, 2004)

Self-Similarity

$$Law(X(\lambda x) \in \mathbb{R}^m; x \in \mathbb{R}^d) = Law(\lambda^H X(x) \in \mathbb{R}^m; x \in \mathbb{R}^d)$$

$$H \in Gl(m, \mathbb{R}), \lambda > 0$$

Wavelet decomposition

$$X(x) = \sum_{\varepsilon,j,k} 2^{jH} \Phi^{\varepsilon}_{j,k}(x) \xi^{\varepsilon}_{j,k}$$

$$\Phi^{\varepsilon} : \mathbb{R}^d \mapsto Gl(m, \mathbb{R}), \Phi^{\varepsilon}_{j,k} = \Phi^{\varepsilon}(2^j x - k), \quad \xi^{\varepsilon}_{j,k} \text{ an iid } N(0, I_m) \; RV$$

to $\overrightarrow{\text{vtdMAP}}$

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square \text{ with:}$$

- X the generated medium or vector field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $\underline{H}_\square$ the Hurst matrix exponent;
- $\underline{F}_\square$ the “topolet” function with matrix values;
- $\xi_\square$ a family of random vectors.

to $\overrightarrow{\text{vtdMAP}}$

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square \text{ with:}$$

- X the generated medium or vector field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $\underline{H}_\square$ the Hurst matrix exponent;
- $\underline{F}_\square$ the “topolet” function with matrix values;
- $\xi_\square$ a family of random vectors.

to $\overrightarrow{\text{vtdMAP}}$

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square \text{ with:}$$

- X the generated medium or vector field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $\underline{H}_\square$ the Hurst matrix exponent;
- $\underline{F}_\square$ the “topolet” function with matrix values;
- $\xi_\square$ a family of random vectors.

to vtd$\overrightarrow{\text{MAP}}$

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_{\square}} \underline{F}_{\square}(2^j x - k)\xi_{\square} \text{ with:}$$

- X the generated medium or vector field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $\underline{H}_{\square}$ the Hurst matrix exponent;
- $\underline{F}_{\square}$ the “topolet” function with matrix values;
- $\xi_{\square}$ a family of random vectors.

to $\overrightarrow{\text{vtdMAP}}$

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square \text{ with:}$$

- X the generated medium or vector field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $\underline{H}_\square$ the Hurst matrix exponent;
- $\underline{F}_\square$ the "topolet" function with matrix values;
- $\xi_\square$ a family of random vectors.

to $\overrightarrow{\text{vtdMAP}}$

$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square$ with:

- X the generated medium or vector field; x the spatial position in the d dimensional domain;
- T_p a percolation tree with a p pruning parameter; $(j,k) \in \mathbb{N} \times \mathbb{Z}^d$ gives the location of each node in T_p;
- $\underline{H}_\square$ the Hurst matrix exponent;
- $\underline{F}_\square$ the "topolet" function with matrix values;
- $\xi_\square$ a family of random vectors.

$\overrightarrow{\text{vtdMAP}}$: Main Features

$\overrightarrow{\text{vtdMAP}}$ possesses some desired features for a realistic vector field model:

- it can generate self-similar random density or vector fields in any range of dimensions;
- some of these parameters are independantly identifiable on the base of mathematical proofs (the morphlet transform can be generalized in a “topolet transform”);
- with some amount of work it can provide density or vector fields with given statistical properties closed to those observed in atmospheric phenomenons.

$\overrightarrow{\text{vtdMAP}}$: Main Features

$\overrightarrow{\text{vtdMAP}}$ possesses some desired features for a realistic vector field model:

- it can generate self-similar random density or vector fields in any range of dimensions;
- some of these parameters are independantly identifiable on the base of mathematical proofs (the morphlet transform can be generalized in a “topolet transform”);
- with some amount of work it can provide density or vector fields with given statistical properties closed to those observed in atmospheric phenomenons.

$\overrightarrow{\text{vtdMAP}}$: Main Features

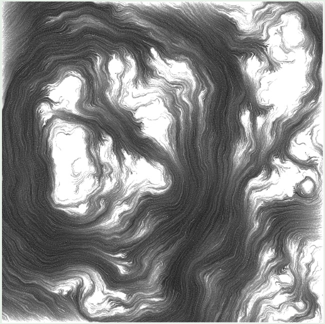
$\overrightarrow{\text{vtdMAP}}$ possesses some desired features for a realistic vector field model:

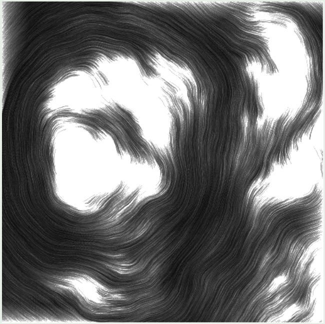
- it can generate self-similar random density or vector fields in any range of dimensions;
- some of these parameters are independantly identifiable on the base of mathematical proofs (the morphlet transform can be generalized in a “topolet transform”);
- with some amount of work it can provide density or vector fields with given statistical properties closed to those observed in atmospheric phenomenons.

Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square$$

Diagonal Hurst Matrix Exponent



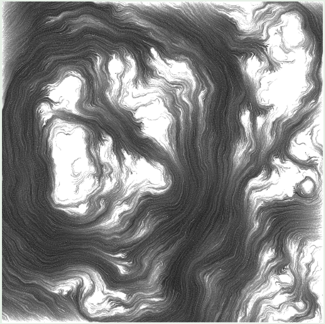


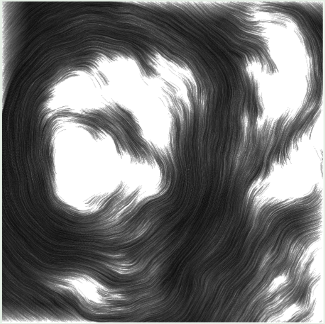
$$\underline{H} = \begin{pmatrix} 1/3 & 0 \\ 0 & 1/3 \end{pmatrix} \qquad \underline{H} = \begin{pmatrix} 2/3 & 0 \\ 0 & 2/3 \end{pmatrix}$$

Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square$$

Diagonal Hurst Matrix Exponent





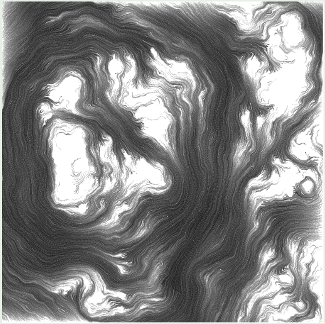
$$\underline{H} = \begin{pmatrix} 1/3 & 0 \\ 0 & 1/3 \end{pmatrix}$$

$$\underline{H} = \begin{pmatrix} \alpha & 0 \\ 0 & \alpha \end{pmatrix}, \ \alpha \in]0,1[$$

Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square$$

Non diagonal Hurst Matrix Exponent

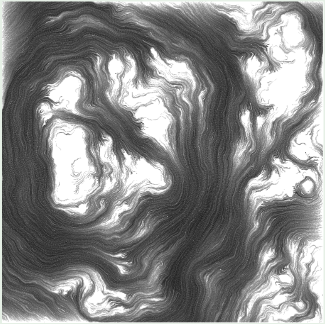


$$\underline{H} = \begin{pmatrix} \alpha & \gamma \\ -\gamma & \alpha \end{pmatrix},\ \alpha = 1/3,\ \gamma \in \mathbb{R}, \lambda_1 = \alpha + i\gamma,\ \lambda_2 = \overline{\lambda_1} = \alpha - i\gamma$$

Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square} \underline{F}_\square(2^j x - k)\xi_\square$$

Non diagonal Hurst Matrix Exponent

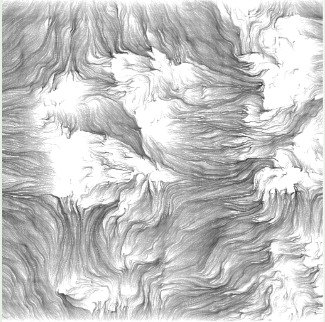


$$\underline{H} = \begin{pmatrix} \alpha & \gamma \\ -\gamma & \alpha \end{pmatrix}, \ \alpha = 1/3, \ \gamma \in \mathbb{R}, \lambda_1 = \alpha + i\gamma, \ \lambda_2 = \overline{\lambda_1} = \alpha - i\gamma$$

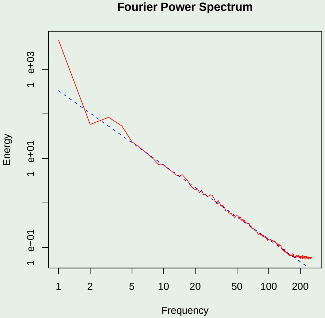
Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_{\square}} F_{\square}(2^j x - k)\xi_{\square}$$

Topolet



Non divergence-free topolet

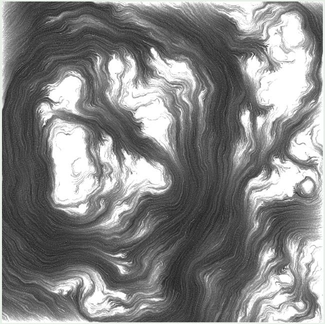


$\beta \approx -1.63$

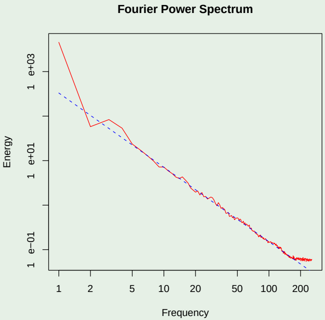
Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-jH_{\square}} F_{\square}(2^j x - k)\xi_{\square}$$

Topolet



Divergence-free topolet

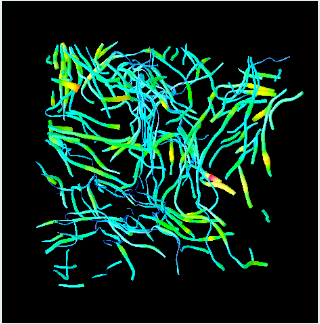


$\beta \approx -1.67$

Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_\square}\, \underline{F}_\square(2^j x - k)\xi_\square$$

Topolet

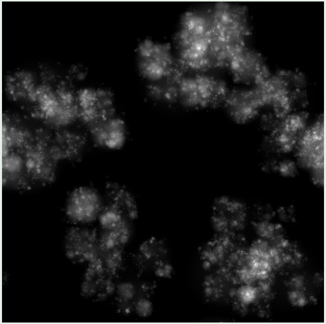


3D divergence-free topolet

Sensivity Analysis

$$X(x) = \sum_{(j,k)\in T_p} 2^{-j\underline{H}_{\square}} \underline{F}_{\square}(2^j x - k)\xi_{\square}$$

2D animation



Outline

- The generation of stratocumulus clouds is well developped;
- The generation of realistic animations is in progress: a deeper sensivity analysis should allow to parameterize animations against real measures.

For Further Reading

- N.Ferlay, H.Isaka, P.Gabriel, A.Benassi. Multi-resolution Analysis Transfer though Inhomogeneous Media PartI, PartII, in revision in Journal of Athmospheric Sciences.
- K.F.Evans,1998: The Spherical Harmonic Discrete Ordinate Method for three-dimensional Athmospheric Radiative Transfer. J. Athmos. Sci., 55, 429-446.
- A.Benassi. Locally Self-Similar Gaussian Processes (Pages 43-54), In Wavelets and Statistics, A.Antoniadis, G.Oppenheim (Editors). Lecture Notes in Statistics 103, Springer-Verlag 1995.
- A.Benassi, S.Jaffard, D.Roux. Elliptics Gaussian Random Proceses. Rev.Mathzmatica Iberoamericana,13(1):19-90,1997.
- C.Bahadoran, A.Benassi, C.Debicki; Operator Self-Similar Gaussian Processes with Stationary Increments. INn Revision in SPA.
- A.Benassi, F.Szczap, A.Davis, M.Masbou,C.Cornet,P.Bleuyard. Thermal Radiative Fluxes through Inhomogeneous Clouds Fields/ a Sensitivity Study using a new Stochastic Cloud Generator. Atmos. Res. Vol. 72, No 1-4,p 291-315 Elsevieur 2004.
- A.Benassi, P.Bleuyard,J.Fayolle. Analysis and Synthesis of Multi-Dimensional Signals Using Morphlets; Application to Sub-Pixel Prediction. In progress.