Chaos, Solitons & Fractals 45 (2012) 246–255

Contents lists available at SciVerse ScienceDirect

Chaos, Solitons & Fractals

Nonlinear Science, and Nonequilibrium and Complex Phenomena

journal homepage: www.elsevier.com/locate/chaos

An algorithm for computing the centered Hausdorff measures of self-similar sets ☆

Marta Llorente [a,*], Manuel Morán [b]

[a] Departamento de Análisis Económico: Economía Cuantitativa, Universidad Autònoma de Madrid, Campus de Cantoblanco, 28049 Madrid, Spain
[b] Departamento Análisis Económico I, Universidad Complutense de Madrid, Campus de Somosaguas, 28223 Madrid, Spain

ARTICLE INFO

Article history:
Received 29 July 2011
Accepted 2 December 2011
Available online 20 January 2012

ABSTRACT

We provide an algorithm for computing the centered Hausdorff measures of self-similar sets satisfying the strong separation condition. We prove the convergence of the algorithm and test its utility on some examples.

1. Introduction

We present an algorithm that takes as input a list $\Psi = \{f_1, f_2, \ldots, f_m\}$ of contracting similitudes in $\mathbb{R}^n$ satisfying the strong separation condition (see Section 2) and gives as output an estimate of the *s-dimensional centered Hausdorff measure*, $C^s(E)$, of the self-similar set E generated by Ψ. Here s is both the similarity and Hausdorff dimension of E and it can be computed from the contracting factors of the similitudes (see (2.1)). To our knowledge this is the first attempt at automatic computation of the exact value of a metric measure (e.g. Hausdorff, packing, spherical, centered, ...), a topic which has generated a considerable quantity of research [1,2,6,7,11,16–22], etc.

The centered Hausdorff measure is a variant of the Hausdorff measure. These measures differ mainly in the natures of the coverings used in their definitions. In the case of the centered Hausdorff measure we consider only covers by closed balls $B(x_i)$ centered at points x_i in the given set.

The standard definition of C^s on subsets of $\mathbb{R}^n$ consists of two steps. Given $A \subset \mathbb{R}^n$, we first compute the premeasure C_0^s as

$$C_0^s A = \lim_{\delta \to 0} \inf \left\{ \sum_{i \in \mathbb{N}} |B_i(x_i)|^s : A \subset \cup B_i \quad \text{and} \quad |B_i| < \delta, x_i \in A \text{ for all } i \right\}. \tag{1.1}$$

Because the suppression of good candidates for the x_i's may cause an increase of the infimum, this premeasure is not monotone, although it is σ-additive (see [13]). In order to avoid this difficulty we define

$$C^s A = \sup \{ C_0^s B : B \subset A, B \text{ closed} \}. \tag{1.2}$$

The set function C^s so obtained is a metric measure. It turns out that the centered Hausdorff measure is bounded by constant multiples of the ordinary Hausdorff measure (see [13]). More precisely

$$2^{-s} C^s(E) \leqslant H^s(E) \leqslant C^s(E)$$

and so the centered Hausdorff dimension and the ordinary Hausdorff dimension coincide. In particular, for the self-similar set E, we have that $0 < C^s(E) < \infty$. Thus C^s is a nice measure, but there remains the question of how to compute $C^s A$ for some subset $A \subset E$. A simplification comes from the following observation: for any Borel set $A \subset \mathbb{R}^n$,

$$C^s A = (C^s E)(\mu A),$$

where is μ the so called *natural* or *empirical probability measure on E*. Therefore, the problem of computing C^s on E reduces to computing $C^s E$. Observe that the μ-measure of any subset $A \subset E$ open in the relative topology ($A = U \cap E$ where U is open) with boundary ∂A having null

☆ This research was supported by the Ministerio de Educación y Ciencia, research project MTM2006-02372.

* Corresponding author. Tel.: +34 91 497 28 77; fax: +34 91 497 29 91.
E-mail addresses: m.llorente@uam.es (M. Llorente), mmoranca@ccee.ucm.es (M. Morán).

0960-0779/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.chaos.2011.12.001

μ-measure (and in particular of any open ball) can be easily obtained with arbitrary accuracy and, hence, the μ-measures of compact subsets of E with μ-null boundaries can also be computed through their complements. This gives a vast class $\mathcal{C}$ of Borel sets with computable μ-measure. In turn, the C^s-measure of any C^s-measurable set can be approximated with arbitrary accuracy by the C^s-measures of closed sets (see [4] Theorem 1.6 b).

Then, how it is possible to compute C^sE? Given the definitions, the task seems out of reach. The first obstruction comes from the need of the second step (1.2) in the definition of C^s. However, in [9] it is proved that, for any subset A of a self-similar set E as above the measure and the premeasure coincide:

Theorem 1.3 (*Theorem 3 in 10*). *Let $A = U \cap E$ where U is either a closed or an open set and E is a self-similar set satisfying the open set condition. Then $C^s_0 A = C^s A$.*

With this result available, C^s seems easier to compute than the Hausdorff or spherical Hausdorff measure. Namely, the differences between these three measures are that, for the Hausdorff measure, one optimizes among coverings by arbitrary convex sets; for the spherical Hausdorff measure one uses arbitrary balls; and for C^s one uses coverings by balls with centers in E. For the Hausdorff and the spherical Hausdorff measures, the classes of available coverings are larger, and hence it is more difficult to find optimal coverings.

The second step which permits the computation of C^s was also given in [9]. There it is proved that computing C^s is equivalent to finding a centered ball with optimal inverse density. For a different approach valid for the strong separation condition see [15] (Theorem 10.1, p.272).

Theorem 1.4 (*Theorem 5 in 10*). *Suppose the invariant set E of the system Ψ satisfies the open set condition, with $\dim_H E = s$ and $|E| = R$, and let μ be the normalized Hausdorff measure on E. Then*

$$C^sE = \inf\left\{\frac{(2d)^s}{\mu(B(x,d))} : x \in E, d > 0\right\}. \tag{1.5}$$

Moreover, if Ψ satisfies the SSC then

$$C^sE = \min\left\{\frac{(2d)^s}{\mu(B(x,d))} : x \in E \quad \text{and} \quad c \leqslant d \leqslant R\right\}, \tag{1.6}$$

where $c := \min_{i,j\in M, i\neq j} \operatorname{dist}(f_iE, f_jE)$ and $R := |E|$.

From now on, $B(x,d)$ denotes the closed radius d ball centered at $x \in \mathbb{R}^n$.

The statement (1.6) is crucial in our approach since it gives that if $B(x,d)$ is a ball of maximal inverse density, then $C^s(E) = \frac{(2d)^s}{\mu(B(x,d))}$. Therefore, to obtain the value of $C^s(E)$ we need only to find an optimal ball and compute its density. This is precisely how the algorithm proceeds. It searches for balls that maximize the density function $h_s(x,d) = \frac{(2d)^s}{\mu(B(x,d))}$. Actually, by *the self-similar tiling principle* (see [12]), we know that finding an optimal ball is equivalent to finding an optimal covering. This is so because we can get an optimal covering tiling the set E with balls of optimal density. E may be tiled, without loss of μ-measure, by tiles similar to a given tile B. By similar we mean that the tile is an image of B under a composition of similitudes in Ψ. The only condition to be imposed on B is that it be closed and have $\mu B > 0$.

Our algorithm computes C^sE and provides an (approximate) ball of maximal density, together with an optimal covering of E by balls centered at E.

Now we describe the main steps of the algorithm. This is done rigorously in Section 3. Recall that the goal is to find the maximal value of $h_s(x,d) = \frac{(2d)^s}{\mu(B(x,d))}$ for $x \in E$ and $d \in [c,R]$ (see (1.6)). For any $k \in \mathbb{N}$, at step k of the algorithm, the set E is replaced by a finite set of points $\{A_k\}$ such that $\overline{\cup A_k} = E$ and the measure μ is replaced by a discrete probability measure μ_k supported on A_k and converging weakly to μ (see (3.4), (3.9) and Lemma 4.1 (iv)). The objective now is to find the maximum of the discrete density function

$$h_k(x, d(x,y)) = \frac{(2d(x,y))^s}{\mu_k(B(x,d(x,y)))} \quad \text{with } x, y \in A_k.$$

Here $d(,)$ stands for the Euclidean distance. For each $x \in A_k$ the algorithm searches for the maximal value of $h_k(x,d(x,y))$ for $y \in A_k$. Once this has been found for every $x \in A_k$, the algorithm finds the maximum of these values with respect to x. Thus we only need to compute exactly $\mu_k(B(x,d(x,y)))$ for every $y \in A_k$. To this end, the points $y \in A_k$ are listed in order of increasing distance to x, and thus the points preceding a given point y in the list always belong to the ball $B(x,d(x,y))$. It is not hard to see that the exact value of $\mu_k(B(x,d(x,y)))$ is obtained from the place of y in this list (see (3.5) for the homogeneous case and (3.11) for the general case). It remains to show that in the limit $k \nearrow \infty$ this process converges to $C^s(E)$. This is done in Section 4. The convergence is shown in two steps. First, by means of the Markov operator M associated to the set E (see Section 2 for notation and definitions), the measures μ_k are shown to converge weakly to the invariant measure μ. The basic properties of these measures yield a sequence of pairs of points $(x_k, y_k) \in A_k \times A_k$ such that $h_k(x_k, d(x_k, y_k)) \to C^s(E)$. However, there is no reason why these (x_k, y_k) should optimize h_k. Nonetheless, we are able to show that this holds asymptotically, which is enough for our purposes. An interesting technical point is that in the proofs an essential role is played by a result of Mattila [10] implying that $\mu(\partial B(x,d)) = 0$.

In Section 5 we apply the algorithm to treat several sets whose centered Hausdorff measures were available in the literature. It is remarkable that in all these cases the optimal value (and also optimal ball and covering) is attained at an early iteration. The algorithm also yields conjectural values (which, in many cases, are upper bounds) for sets whose measure is unknown. Research is in progress to explore the rate of convergence and show that the method yields the precise values of $C^s(E)$. Preliminary results seem to indicate that, for self-similar sets with less than four similarities, four decimal digits of accuracy are attainable by personal computers without any serious effort to optimize the code's design.