USING FRACTAL GEOMETRY FOR SOLVING DIVIDE-AND-CONQUER RECURRENCES

SIMANT DUBE[1]

Abstract

A relationship between the fractal geometry and the analysis of recursive (divide-and-conquer) algorithms is investigated. It is shown that the dynamic structure of a recursive algorithm which might call other algorithms in a mutually recursive fashion can be geometrically captured as a fractal (self-similar) image. This fractal image is defined as the attractor of a mutually recursive function system. It then turns out that the Hausdorff-Besicovich dimension D of such an image is precisely the exponent in the time complexity of the algorithm being modelled. That is, if the Hausdorff D-dimensional measure of the image is finite then it serves as the constant of proportionality and the time complexity is of the form $\Theta(n^D)$, else it implies that the time complexity is of the form $\Theta(n^D \log^p n)$, where p is an easily determined constant.

1. Introduction

The analysis of the time complexity of algorithms is of fundamental importance to computer scientists. A great number of useful algorithms use the divide-and-conquer approach, in which the original problem is reduced to a number of smaller problems [1, 6]. In this paper, we consider a new fractal geometry based approach to analyze such algorithms, in which the size of a smaller problem is related to that of the original problem by a multiplicative factor.

The problem of analysis of such recursive algorithms reduces to solving divide-and-conquer recurrence relations. A number of methods have been developed for solving such recurrence relations, and also for general recurrence relations [11, 12, 16].

In [6] the Master method to solve divide-and-conquer recurrences is discussed. The Master Method is based on the Master Theorem, which is adapted from [5]. In past literature, mutual recurrence relations of more general nature have been considered. In [16], such recurrence relations are called multi-dimensional linear first order recur-

[1] Iterated Systems Inc., Seven Piedmont Centre, Atlanta GA 30305, USA.

rences. Divide-and-conquer recurrences are called extended first order recurrences. A divide-and-conquer recurrence can be reduced to a (single-dimensional) linear first order recurrence and a secondary recurrence [16].

In this paper, we present a new approach to solve mutual divide-and-conquer recurrences, which gives more general results in a simpler manner.

Surprisingly, we make use of the recent developments in fractal geometry, which has gained a remarkable popularity among scientists and mathematicians since it was shown by Mandelbrot in [14] that many natural objects possess fractal (self-similar) geometries. If one magnifies one of the parts of a self-similar object then it resembles the whole. Clouds, mountains, trees, human circulatory system are examples of fractal objects.

An important step in the development of "computational fractal geometry" is taken by Barnsley in [2]. He has developed the theory of Iterated Function Systems (IFS), originally introduced by Hutchinson in [13]. For image generation and compression purposes, IFS are generalized to Mutually Recursive Function Systems (MRFS) in [7] and are also studied in [9]. MRFS are related to Recurrent IFS, introduced in [3]. An interesting special case of MRFS is studied in [8].

An MRFS consists of n components (images) defined in a mutually recursive fashion as unions of each others under affine transformations. IFS is a special case of MRFS when $n = 1$.

Results on Hausdorff-Besicovich dimension of objects defined by MRFS are shown in [15], which generalize those in [3]. In this paper, we will be using the results from [15] to build a relationship between fractal geometry and analysis of recursive algorithms.

At a conceptual level, the notion of self-similarity is not limited solely to images but can be used to describe many natural phenomena like distribution of noise on a channel, Brownian motion of particles in air [4]. In this paper, we show that a divide-and-conquer algorithm is also "self-similar" as it is made of its smaller "copies". Here self-similarity is temporal while in case of a natural object it is spatial.

```
Algorithm A(B[1 ... n])
Array B;
if n = 1 then ..., print("hello");
    else call A(B[1 ... n/2]),
         call A(B[n/4 ... 3n/4]),
         call A(B[n/2 ... n]),
end A;
```
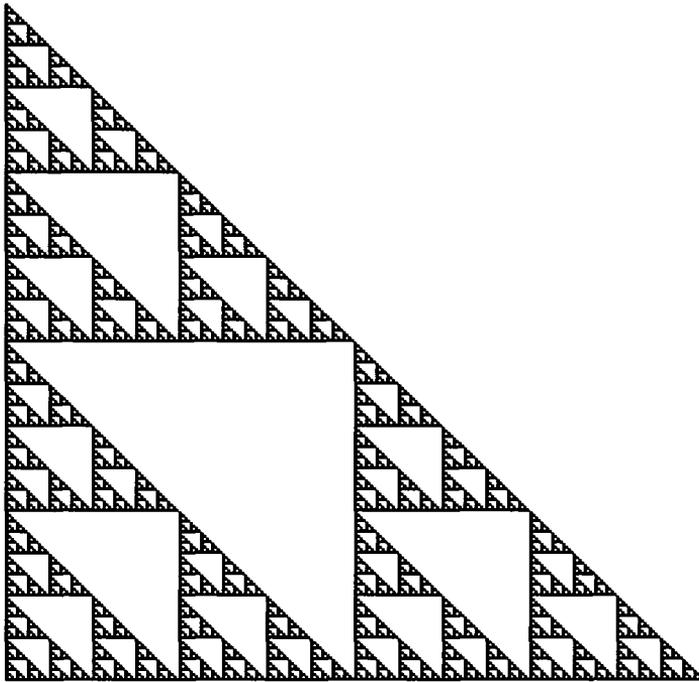
FIGURE 1. A recursive algorithm.

FIGURE 2. Dynamic structure of the algorithm in Figure 1 captured by the Sierpinski Triangle.

For example, consider a recursive algorithm A as shown in Figure 1. The algorithm A calls itself 3 times and at each recursive call the input size is halved. The system of recurrence relations is

$$T(n) = 3T(n/2),$$
$$T(1) = O(1),$$

where $T(n)$ denotes the time complexity of the algorithm on input of size n. Note that we assume that all computation is done at the "trivial-case" $n = 1$. Therefore $T(n) = \Theta(n^{\log_2 3})$.

Now A can be modelled by an IFS on 2-D Euclidean space and having the following three transformations, each with contractivity 0.5,

$$w_1(x, y) = (0.5x, 0.5y),$$
$$w_2(x, y) = (0.5x + 0.5, 0.5y),$$
$$w_3(x, y) = (0.5x, 0.5y + 0.5),$$

which map the unit square $U = [0, 1]^2$ into the lower-left, lower-right and upper-left quadrants, respectively. The attractor of the IFS is the well-known Sierpinski Triangle; see Figure 2. Its fractal dimension is $\log_2 3$. This is no coincidence as can be intuitively deduced as follows:

Consider the intuitive definition of fractal dimension of a self-similar image O which implies that if O has fractal dimension D then

$$\text{(number of self-similar copies)} \approx C \text{(magnification factor)}^D, \tag{1}$$

where C is some positive constant.

Now consider the recursive algorithm A such that at each of its recursive calls the size of the input is reduced by a multiplicative factor. Each such calls creates a "copy" of A on smaller input. Since we assume that the only computation is done at the "trivial-case" when the size of the input is 1, the total time taken by A on an input of size n is the total number of recursive calls made with input size equal to 1. How many such trivial-case recursive calls are made? For this, we rewrite (1) as

$$\text{(number of recursive calls)} \approx C \text{(magnification factor)}^D. \tag{2}$$

In our case

$$\text{magnification factor} = \frac{\text{size of the original input}}{\text{size of the trivial-case input}} = \frac{n}{1} = n.$$

Therefore, from (2) the time complexity of the algorithm A is $T(n) \approx Cn^D$.

Now this interrelationship between divide-and-conquer recurrences and fractals can be generalized to a system of recurrences and in which the multiplicative factor is any real number. Also, one can easily handle the case in which computation is done at other recursion levels besides the trivial-case. This generalization is the aim of this paper.

Consider a group A of n mutually recursive algorithms and let one algorithm be distinguished as the main algorithm (main "routine" in the terminology of programming languages) which is called first. An algorithm may call itself or any other algorithm. In such a recursive call, the size of input is reduced by a multiplicative factor.

Now A can be modelled as an MRFS M with n components, such that the execution of A corresponds "graphically" with the sequence of images generated while executing the Deterministic Algorithm on M. Let the attractor of M (the fractal image defined by M) be O. Then the main theorem states the mathematical relationship between the Hausdorff-Besicovich dimension D and the Hausdorff D-dimensional measure of O and the time complexity of A. A number of remarks can be made upon these results:

(1) It provides a mathematically rigorous method to analyze mutually recursive algorithms. It generalizes the known methods to solve recurrence relations, for example, the Master Theorem as given in [6] is a special case of the main result of this paper.

(2) The approach in this paper is general as it circumvents any *discrete* analysis as has been done in past literature (for example, in [5, 6, 11, 12, 16]) but instead uses already known sophisticated results from the mathematics of fractal geometry [2, 3, 15]. It provides a pleasing link between discrete mathematics and continuous mathematics.

2. Preliminaries

2.1. Euclidean spaces and contractive mappings   Throughout this paper, (R^n, Euclidean distance), the n-dimensional real space with the Euclidean metric is the underlying complete metric space that is $X = R^n$ for some integer $n \geq 1$.

A 2-dimensional *affine transformation* $w : R^2 \to R^2$ is defined by

$$w \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_{11}x + a_{12}y + b_1 \\ a_{21}x + a_{22}y + b_2 \end{bmatrix},$$

where a_{ij}'s and b_i's are real constants [2]. Similarly, a 1-dimensional affine transformation $w : R \to R$ is defined by $w(x) = ax + b$, where a and b are real constants. Likewise, we can define an affine transformation on R^n for all integers $n > 2$. In this paper, one can restrict oneself to those affine transformations which only scale and translate (that is, no rotation).

A transformation $f : X \to X$ on a metric space (X, d) is called a *contractive mapping* if there is a constant $0 \leq s < 1$ such that

$$d(f(x), f(y)) \leq s.d(x, y) \text{ for all } x, y \in X.$$

Any such number s is called a *contractivity factor* of f. If f satisfies the condition

$$d(f(x), f(y)) \leq s.d(x, y) \text{ for all } x, y \in X,$$

where $s > 0$, then f is called a *similitude*.

2.2. Fractal dimension   Let A be an "image" in $X = R^n$, that is, A is a nonempty compact subset of X. The set of all images in X is denoted by $\mathscr{H}(X)$. Let $\epsilon > 0$. Let $B(x, \epsilon)$ denote the closed ball of radius ϵ and centre at a point $x \in X$. That is,

$$B(x, \epsilon) = \{y \in X | d(x, y) \leq \epsilon\}.$$

Let $\mathcal{N}(A, \epsilon)$ be the least number of closed balls of radius ϵ needed to cover A. That is,

$$\mathcal{N}(A, \epsilon) = \text{smallest integer } M \text{ such that } A \subseteq \bigcup_{n=1}^{M} B(x_n, \epsilon),$$

for some set of distinct points $\{x_n | n = 1, 2, \ldots, M\} \subseteq X$.

Let $f(\epsilon)$ and $g(\epsilon)$ be real valued functions of the positive real variable ϵ. Then $f(\epsilon) \approx g(\epsilon)$ means that

$$\lim_{\epsilon \to 0} \frac{\log(f(\epsilon))}{\log(g(\epsilon))} = 1.$$

The intuitive idea behind the definition of fractal dimension is that a set A has fractal dimension D if $\mathcal{N}(A, \epsilon) \approx C\epsilon^{-D}$ for some positive constant C. Mathematically we define it as the limit

$$\lim_{\epsilon \to 0} \frac{\log \mathcal{N}(A, \epsilon)}{\log(1/\epsilon)},$$

if it exists.

Another way to define fractal dimension is using the "Box-counting", that is, counting the numbers of boxes of a grid overlaying the image, see [2].

2.3. Hausdorff-Besicovich dimension The Hausdorff-Besicovich fractal dimension of a set $A \in \mathcal{H}(X)$ is a dimensional index of A. Define the *diameter* of A as

$$\text{diam}(A) = \sup\{d(x, y) | x, y \in A\}.$$

Let $0 < \epsilon < \infty$ and $0 \le p \le \infty$. Let $\mathcal{A}$ denote the set of sequences of subsets $\{A_i \subset A\}$ such that $A = \cup_{i=1}^{\infty} A_i$. That is, each element of $\mathcal{A}$ is a "covering" of A. Then we define a real number describing each covering $\{A_i\} = \{A_1, A_2, \ldots\} \in \mathcal{A}$ of A,

$$\mathcal{C}(\{A_i\}, p) = \sum_{i=1}^{\infty} (\text{diam}(A_i))^p.$$

In the above we use the convention that $(\text{diam}(A_i))^0 = 0$ when A_i is empty. We consider the infimum of the above,

$$\mathcal{M}(A, p, \epsilon) = \inf\{\mathcal{C}(\{A_i\}, p) | \{A_i\} \in \mathcal{A}, \text{ diam}(A_i) < \epsilon, \text{ for } i = 1, 2, \ldots\}.$$

We now define the *Hausdorff p-dimensional measure* of A as

$$\mathcal{M}(A, p) = \sup\{\mathcal{M}(A, p, \epsilon) | \epsilon > 0\}.$$

Since $\mathcal{M}(A, p, \epsilon)$ is a nonincreasing function of ϵ, one has to consider "finer" coverings of A to estimate $\mathcal{M}(A, p)$.

There is unique real number D_H which is less than or equal to the dimension of the underlying Euclidean space such that

$$\mathcal{M}(A, p) = \begin{cases} \infty & \text{if } p < D_H \text{ and } p \in [0, \infty), \\ 0 & \text{if } p > D_H \text{ and } p \in [0, \infty). \end{cases}$$

The real number D_H is called the *Hausdorff-Besicovich dimension* of the set A [2]. However for the images defined by MRFS considered in this paper, the fractal dimension and Hausdorff-Besicovich dimension are always equal. This is because these MRFS are "nonoverlapping". As for nonoverlapping IFS these two dimensions are always same [2], similarly they are same for nonoverlapping MRFS [3, 15].

Hausdorff-Besicovich dimension and Hausdorff p-dimensional measure can be used to compare the "sizes" of two fractals A_1 and A_2. Let A_1 and A_2 have Hausdorff-Besicovich dimensions D_1 and D_2 respectively. Then A_1 is bigger than A_2 if and only if D_1 is greater than D_2. If $D_1 = D_2$ then we compare Hausdorff D_1-dimensional measure C_1 of A_1 and Hausdorff D_2-dimensional measure C_2 of A_2. Then A_1 is bigger than A_2 if and only if C_1 is greater than C_2.

2.4. Asymptotic notations   We have two cases depending on the domain of the function. A function $f(n)$ is said to be a function of *large* reals if n takes values from the real interval $[1, \infty]$. A function $f(\epsilon)$ is said to be a function of *small* reals if ϵ takes values from the real interval $[0, 1]$.

> **Algorithm** A (input of size $= n$)
> **if** $n = 1$ **then** {do trivial things};
>   **else**
>   {call other algorithms recursively},
>   {maybe do some additional computation},
>   **end** A;

FIGURE 3. Structure of a recursive algorithm.

In this paper, the time complexity $T(n)$ of an algorithm on input of size n will be a function of large reals (we will show that the input size can be treated as real instead of positive integer). The function $\mathcal{N}(O, \epsilon)$, the least number of ϵ-balls needed to cover an image O, will be a function of small reals.

For functions of large reals, we have the same standard notations and definitions of asymptotic bounds as are used for functions of natural numbers [6].

We generalize the definitions of asymptotic bounds to functions of small reals but with one important difference – for functions of real variable ϵ, the asymptotic case corresponds when ϵ becomes sufficiently close to 0.

For functions of small reals, we denote the asymptotic upper, lower and tight bounds by O_R, Ω_R and Θ_R respectively.

3. Divide-and-conquer algorithms

Consider a group $A = \{A_1, A_2, \ldots, A_n\}$ of N algorithms. One algorithm is called first and therefore is distinguished as the *main algorithm*. The time complexity of A is therefore the time complexity of the main algorithm. We assume that there are no unreachable algorithms in A, that is, it should be possible to call every algorithm during the execution of A on some input.

The structure of each $A_i \in A$, illustrated in Figure 3, is as follows.

1. The algorithm A_i can call any algorithm $B \in A$ such that the size of the input is changed by a multiplicative factor. If the size of original input is n and the size of the input in a recursive call is changed to n/b where $1/b$ is the multiplicative factor, then we interpret n/b as either $\lceil n/b \rceil$ or as $\lfloor n/b \rfloor$. This is because the size of the input must be an integer. (Here $\lceil x \rceil$ and $\lfloor x \rfloor$ denote respectively the least integer not less than x and the greatest integer not exceeding x.)
2. The algorithm A_i can do some additional computation taking $\Theta(n^D)$ steps, where D is a nonnegative real number. If $D = 0$ then this computation can be ignored.
3. The algorithm A_i performs a constant amount of computation at the trivial case, when the input size is equal to 1.

If there is a chain of recursive calls $B_1, B_2, \ldots, B_m$ where $B_i \in A$, $i = 1, 2, \ldots, n$, B_i calls B_{i+1} and $B_1 = B_m$, then the size of the input in call to B_m should be strictly less than the size of the input in call to B_1. In other words, over every possible loop, the size of the input is "contracted" by a multiplicative factor.

We need to consider a technicality. Suppose in a recursive call, the size n of the input is contracted by a factor of $1/b$. It is possible that b does not divide n. Since input size can be only integer, we assumed that we interpret n/b as either $\lceil n/b \rceil$ or as $\lfloor n/b \rfloor$.

But what if we allow the input size to be a real number?

In this case, let the trivial-case in Figure 3 occur when the input size $n \leq 1$. We now show that this assumption does *not* affect the analysis of the time complexity.

LEMMA 1. *Let $T(n)$ be the time complexity of a group of recursive algorithms A when the input size is required to be an integer. Let $T'(n)$ be its time complexity in the hypothetical situation where the input size is allowed to be a positive real number and the trivial-case occurs if the input size is less than or equal to 1. Then $T'(n) = \Theta(T(n))$.*

PROOF. The proof is a straightforward generalization of a similar result in [6]. It follows from the fact that calls are contractive over loops and the inequalities $\lceil x \rceil \leq x + 1$ and $\lfloor x \rfloor \geq x - 1$.

Therefore we will now adopt the "real-number" model and let the input size be real numbers.

EXAMPLE 1 Consider Figure 4 which shows a group of mutually recursive algorithms. One algorithm is distinguished as the main routine. The main routine and other "subroutines" call each other recursively. In total, there are four other subroutines. If a routine calls another on input size n/b then there is an appropriately labeled arc from the called routine to the calling routine. The "additional" computation performed by the routines is indicated by nodes drawn as squares. Note that each of the subroutines 2 and 4 perform $\Theta(n^2)$ additional computation and the subroutine 3 performs $\Theta(n)$ additional computations.
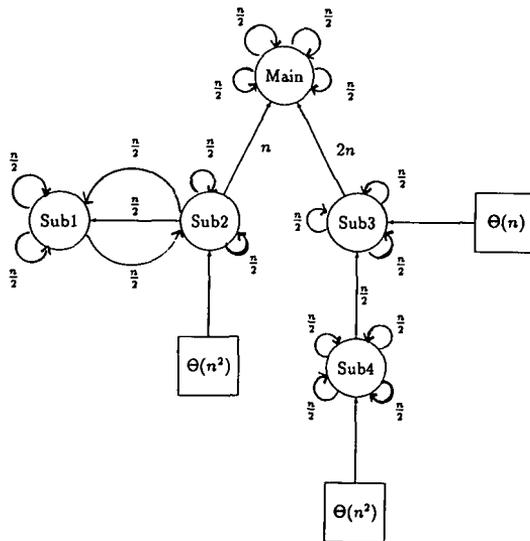

FIGURE 4. A group of mutually recursive algorithms.

4. Mutually recursive function systems

Let $X = R^k$, the k-dimensional Euclidean space, be the underlying metric space.

Now we generalize Mutually Recursive Function Systems (MRFS) as studied in [7, 9] or Geometric Graph Directed Construction in [15], to condensation MRFS just to facilitate our discussion.

A *condensation MRFS M* is a quadruple (V, C, W, G) where:

(a) $V = \{V_1, V_2, \ldots, V_n\}$, $n \geq 1$ is a finite set of nonempty compact subsets of X,

(b) $C = \{C_1, C_2, \ldots, C_m\}$, $m \geq 0$ is a finite set of *condensation sets*, that is, nonempty compact subsets of X each of which has fractal dimension,

(c) G is a directed graph with vertex set consisting of integers $1, \ldots, n + m$ and similitudes $w_{i,j}$ of X where $(i, j) \in G$ with contractivity factor $s_{i,j}$ such that

   1. for each i, $1 \leq i \leq n$, there is some j such that $(i, j) \in G$,
   2. for each i, $1 \leq i \leq n$, if there is some j, $n < j \leq n + m$ such that $(i, j) \in G$ then $w_{i,j}$ is the identity map,
   3. for each i, $n \leq i \leq n + m$, $w_{i,i}$ is the identity map and $(i, j) \in G$ if and only if $i = j$, and
   4. if the path component of G rooted at the vertex i_1, $1 \leq i_1 \leq n$ is a cycle, $[i_1, \ldots, i_q, i_{q+1} = i_1]$, then $\prod_{k=1}^{q} s_{i_k, i_{k+1}} < 1$.

(d) The graph G defines a natural mapping W which maps the space $\mathcal{H}(X)^{n+m}$, the vectors of length $n + m$ of compact subsets of X, to itself,

$$W : \mathcal{H}(X)^{n+m} \to \mathcal{H}(X)^{n+m}$$

such that if

$$W(A_1, A_2, \ldots, A_{n+m}) = (B_1, B_2, \ldots, B_{n+m}),$$

then for each i,

$$B_i = \cup_{j=1}^{n+m} w_{i,j}(A_j).$$

Therefore W is a matrix of similitudes such that $W_{i,j} = w_{i,j}$ the similitude which contributes to component i from component j and the zero function for the rest. The *attractor* of the MRFS is obtained by iterating W starting with the vector (V, C)

$$(K_1, \ldots, K_{n+m}) = \lim_{n \to \infty} W^{(n)}(V_1, \ldots, V_n, C_1, \ldots, C_m)$$

and then defining the attractor O as

$$O = \cup_{i=1}^{n+m} K_i,$$

where clearly $K_{n+1} = C_1, \ldots, K_{n+m} = C_m$. This iteration method to approximate the attractor is called the *Deterministic Algorithm* for generation of fractal images. The set O is the unique fixed point of W when the last m components are fixed to be the condensation sets C.

In this paper, we restrict ourselves to nonoverlapping MRFS. M is called *nonoverlapping* if each i, $1 \leq i \leq n$,

$$\{w_{i,j}(K_j)|(i, j) \in G\}$$

is a nonoverlapping family (that is, it is either totally disconnected or just-touching and therefore satisfies the open set condition introduced by Hutchinson, see [2, 13]). A condensation MRFS $M = (V, C, W, G)$ is called simply as an MRFS if $C = \phi$. For many interesting fractal images generated by MRFS see [7, 9].

5. The Mauldin-Williams Theorem

In this paper, for the sake of completeness, we present already known results on Hausdorff-Besicovich dimension of attractors of nonoverlapping MRFS.

LEMMA 2. *Let $M = (V, C, W, G)$ be a nonoverlapping condensation MRFS. Let O be the attractor of M. Let the the Hausdorff-Besicovich dimension of O be α. If the Hausdorff α-dimensional measure k of O is finite then $\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-\alpha})$.*

PROOF. If M is nonoverlapping, then there exists a real number $\epsilon_0 > 0$ such that for all $\epsilon \leq \epsilon_0$ when, in the definition of fractal dimension, O is covered with $\mathcal{N}(O, \epsilon)$ balls of radius ϵ, then these balls are nonoverlapping. In this case the covering, for which infimum in the definition of Hausdorff-Besicovich dimension is achieved, is the one found by letting each A_i in the covering $\{A_i\}$ be a subset of one of these $\mathcal{N}(O, \epsilon)$ balls. Therefore, for all $\epsilon \leq \epsilon_0$, $\mathcal{M}(O, \alpha, \epsilon) = \mathcal{N}(O, \epsilon)\epsilon^{\alpha}$. Let $D' = \text{diam}(O)$. Since $\mathcal{M}(O, \alpha, \epsilon)$ is a nonincreasing function of ϵ, and since $k = \mathcal{M}(O, \alpha) = \sup\{\mathcal{M}(O, \alpha, \epsilon)|\epsilon > 0\}$, thus for all $\epsilon > 0$,

$$\mathcal{M}(O, \alpha, D')\epsilon^{-\alpha} \leq \mathcal{N}(O, \epsilon) \leq k\epsilon^{-\alpha}. \tag{3}$$

Therefore, if k is finite, then $\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-\alpha})$.

In the above proof, if k is finite then from (3) and the fact that $\mathcal{M}(O, \alpha, \epsilon)$ is a nonincreasing function of ϵ, k can be considered as the constant of proportionality in

$$\mathcal{N}(O, \epsilon) \propto (\epsilon^{-\alpha})$$

for the asymptotic case when ϵ is sufficiently small. Later we will see in Theorem 6 that if k is infinite, then $\mathcal{N}(O, \epsilon) = \Theta(\epsilon^{-\alpha} \log^p \epsilon^{-1})$, which incidentally proves the known fact that for nonoverlapping MRFS, fractal dimension and Hausdorff-Besicovich dimension are always same. For this just check that

$$\alpha = \lim_{\epsilon \to 0} \frac{\log \mathcal{N}(O, \epsilon)}{\log(1/\epsilon)}.$$

Let $M = (V, \phi, W, G)$ be an MRFS where $|V| = n$. Define an $n \times n$ matrix $S = (s_{i,j})$ where $s_{i,j}$ is the contractivity factor of similitude $W_{i,j}$. If $(i, j) \notin G$ then $s_{i,j} = 0$. For any nonnegative real β, define S_β as $s_{\beta;i,j} = s_{i,j}^\beta$. Let $\Phi(\beta)$ be the modulus of the largest eigenvalue of S_β, which is also its spectral radius from the Perron-Frobenius Theorem [10].

A digraph is called strongly connected if there exists a directed path between any of its two nodes. A strongly connected component of a digraph G is a maximal subgraph H of G such that H is strongly connected. Thus, the strongly connected components of G are pairwise disjoint. It is possible that they do not cover G. It is also possible for such a component to consist of a single vertex looped on itself. A vertex is not considered to be strongly connected unless it is looped on itself.

THEOREM 1 *Let $M = (V, \phi, W, G)$ be an MRFS and let G be strongly connected. Then the Hausdorff-Besicovich dimension of the attractor of M is the nonnegative real number α such that $\Phi(\alpha) = 1$.*

PROOF. This can be found in [3].

Given a digraph G, let $SC(G)$ denote the set of strongly connected components of G. Also for a strongly connected component $H \in SC(G)$, let α_H denote the Hausdorff-Besicovich dimension of the attractor of the MRFS whose underlying digraph is H.

Now we state the generalization of Theorem 1.

THEOREM 2 [The Mauldin-Williams Theorem] *Let $M = (V, \phi, W, G)$ be a non-overlapping MRFS. Then the Hausdorff-Besicovich dimension of the attractor of M is given by*

$$\alpha = \max\{\alpha_H | H \in SC(G)\}$$

and its Hausdorff α-dimensional measure is finite if and only if between any two elements of

$$\{H \in SC(G) | \alpha_H = \alpha\}$$

there exists no path in G.

PROOF. This can be found in [15].

The definition of condensation MRFS can be generalized to the case when a component j can contribute multiply to another component i. Then there can be multiple arcs from one vertex to another in the underlying digraph. However such a condensation MRFS can be still simulated by another one in which there is at most one arc from one vertex to another.

THEOREM 3 *Let M be a condensation MRFS in which a component can contribute multiply to another component under different similitudes. Then there exists an equivalent condensation MRFS with the same attractor in which a component can contribute to another only once under a similitude, that is, there is at most one arc from one vertex to another in the underlying digraph.*

PROOF. The proof is simple and can be found in [7] where the equivalence of MRFS and recurrent IFS defined in [3] is proved.

Considering such generalized MRFS, two interesting special cases are as follows.

COROLLARY 1 *Consider a nonoverlapping IFS with N similitudes $w_1, w_2, \ldots, w_N$. Let the contractivity factors of $w_1, w_2, \ldots, w_N$ be $s_1, s_2, \ldots, s_N$ respectively. Then the Hausdorff-Besicovich dimension of the attractor of the IFS is the nonnegative unique real number satisfying the equation*

$$s_1^D + s_2^D + \cdots + s_N^D = 1.$$

Note that the IFS is equivalent to generalized MRFS $(\{V\}, \phi, W, G)$, where G is a single node with N self-loops and $W : \mathcal{H}(X) \to \mathcal{H}(X)$ is

$$W(A) = w_1(A) \cup w_2(A) \cup \ldots \cup w_N(A).$$

COROLLARY 2 *Suppose M is a condensation MRFS in which there can be mutiple arcs from one vertex to another in its underlying digraph G. Suppose each of the similitudes labeling an arc in G has a contractivity factor equal to s, where s is real number in the open real unit interval $(0, 1)$. Let λ be the eigenvalue of the maximum modulus of the connection matrix C where $C_{i,j}$ is the number of arcs from vertex i to vertex j. Then the Hausdorff-Besicovich dimension of the attractor of M is $\log_b |\lambda|$, where $b = 1/s$.*

These special cases can be derived from the Mauldin-Williams Theorem or found in [2, 13]. In [17], the case 2 is proved for an integer b.

EXAMPLE 2 Consider a nonoverlapping MRFS with components A, B and C and the contribution of corresponding components to others shown by the following mutually recursive definitions:

$$A = w_1(A) \cup w_2(A) \cup w_3(A),$$
$$S = id(A) \cup w_4(C),$$
$$C = w_5(B) \cup w_6(C).$$

Let the contractivity factors of w_i be $\frac{1}{2}$ if $i \neq 3$ and of w_3 be $\frac{1}{4}$. Now the underlying graph has two strongly connected components – H_1 consisting of node A and H_2 consisting of B and C. From Corollary 1, α_{H_1} is the positive real number D satisfying

$$\left(\tfrac{1}{2}\right)^D + \left(\tfrac{1}{2}\right)^D + \left(\tfrac{1}{4}\right)^D = 1,$$

and therefore it is $\log_2(\sqrt{2}+1)$. To compute α_{H_2}, we consider the connection matrix

$$C = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}.$$

The maximum magnitude eigenvalue is $(\sqrt{5}+1)/2$ and therefore $\alpha_{H_2} = \log_2((\sqrt{5}+1)/2)$ from Corollary 2. Thus the fractal dimension of the attractor of the MRFS is α_{H_1} since $\alpha_{H_1} > \alpha_{H_2}$. Also the Hausdorff D-dimensional measure is finite where $D = \alpha_{H_1}$.

Theorem 2 also holds for condensation MRFS in the following obvious sense. In the underlying digraph of a condensation MRFS, each condensation set with its single self-loop is a strongly connected component. Therefore, if C is a condensation set with Hausdorff-Besicovich dimension equal to D, then we have a strongly connected component H containing C such that $\alpha_H = D$.

Characterizing $\mathcal{N}(O, \epsilon)$

We now use the results on fractal dimension to rephrase them in asymptotic notation Θ_R for small reals (see preliminaries). From now onwards, we will assume that ϵ is a nonnegative real variable taking values in the unit interval $[0, 1]$. The goal is to use the Mauldin-Williams Theorem to characterize $\mathcal{N}(O, \epsilon)$ in terms of ϵ.

We first prove the results for a condensation IFS (that is an MRFS with a single component and a single condensation set).

THEOREM 4 *Consider a condensation IFS:*

$$A = w_1(A) \cup w_2(A) \cup \ldots \cup w_N(A) \cup C,$$

and let its attractor be O and let the Hausdorff-Besicovich dimension of the condensation set C be D_1. Let the Hausdorff-Besicovich dimension of the attractor O' of the IFS

$$A = w_1(A) \cup w_2(A) \cup \ldots \cup w_N(A)$$

be D_2. Then if $D_1 \neq D_2$,

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-\max\{D_1, D_2\}});$$

otherwise if $D_1 = D_2 = D$,

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D} \log \epsilon^{-1}).$$

PROOF. The condensation IFS has two strongly connected components, one having condensation set C and the other having set A. The case $D_1 \neq D_2$ follows from the Mauldin-Williams Theorem and Lemma 2.

For $D_1 = D_2$ we prove the theorem. Let $W : \mathcal{H}(X) \to \mathcal{H}(X)$ be the mapping defined as

$$W(A) = w_1(A) \cup w_2(A) \cup \ldots \cup w_N(A)$$

for all $A \in \mathcal{H}(X)$. W is a contractive mapping on the complete metric space $(\mathcal{H}(X), h)$ where h is the Hausdorff metric and its unique fixed point is O' and its contractivity factor is $s = \max\{s_1, s_2, \ldots, s_N\}$, where s_i is the contractivity factor of transformation w_i [2].

One quickly verifies that the attractor O of the condensation IFS is

$$O = O' \cup C \cup W(C) \cup W^2(C) \cup W^3(C) \cup \ldots \tag{4}$$

Note that O' is actually $\lim_{n \to \infty} W^n(C)$. To see why the above is true, execute the Deterministic Algorithm. Clearly, $W^i(C)$ for all $i \geq 0$ has dimension D as dimension is preserved under similitudes. In fact, one can show that there exist two positive constants k_1 and k_2 such that for all i,

$$k_1 \epsilon^{-D} \leq \mathcal{N}(W^i(C), \epsilon) \leq k_2 \epsilon^{-D}. \tag{5}$$

We show this by induction on i. The basis is true since $W^0(C)$ is C, whose dimension is D. For the inductive step note that

$$W^{i+1}(C) = \bigcup_{j=1}^{N} w_j(W^i(C)).$$

Since IFS is nonoverlapping this implies that (see [2])

$$\mathcal{N}(W^{i+1}(C), \epsilon) = \sum_{j=1}^{N} \mathcal{N}\left(W^i(C), \frac{\epsilon}{s_j}\right).$$

From the induction hypothesis for i, for all j,

$$k_1 \left(\frac{\epsilon}{s_j}\right)^{-D} \leq \mathcal{N}\left(W^i(C), \frac{\epsilon}{s_j}\right) \leq k_2 \left(\frac{\epsilon}{s_j}\right)^{-D}$$

and therefore

$$k_1 \sum_{j=1}^{N} \left(\frac{\epsilon}{s_j}\right)^{-D} \leq \mathcal{N}(W^{i+1}(C), \epsilon) \leq k_2 \sum_{j=1}^{N} \left(\frac{\epsilon}{s_j}\right)^{-D}.$$

Hence

$$k_1 \epsilon^{-D} \sum_{j=1}^{N} s_j^D \leq \mathcal{N}(W^{i+1}(C), \epsilon) \leq k_2 \epsilon^{-D} \sum_{j=1}^{N} s_j^D,$$

and since $s_1^D + s_2^D + \cdots + s_N^D = 1$, we finally get

$$k_1 \epsilon^{-D} \leq \mathcal{N}(W^{i+1}(C), \epsilon) \leq k_2 \epsilon^{-D}.$$

Now $C, W(C), W^2(C), \ldots$ is a Cauchy sequence and for any i, there exists a positive constant k such that

$$h(W^i(C), W^j(C)) \leq k s^i \text{ for all } j > i,$$

where s is the contractivity factor of W and $k = h(C, W(C))/(1-s)$. Thus if $\epsilon > cs^i$, where c depends on k and the underlying metric space $X = R^n$, and if we are covering $C, W(C), \ldots, W^i(C)$ by ϵ-balls then $W^{i+1}(C), W^{i+2}(C), \ldots$ also get covered. Thus for any ϵ, the smallest such i has to satisfy

$$i = c_1 \log_s \epsilon, \text{ for some positive constant } c_1,$$

and since $s < 1$, therefore for the natural logarithmic function,

$$i = c_2 \log \epsilon^{-1} \text{ for some positive constant } c_2.$$

From (4), the number of ϵ-balls needed to cover O is

$$\begin{aligned}
\mathcal{N}(O, \epsilon) &= \Theta_R(\mathcal{N}(O', \epsilon) + \mathcal{N}(C \cup W(C) \cup W^2(C) \cup W^3(C) \cup \ldots, \epsilon)) \\
&= \Theta_R(\mathcal{N}(O', \epsilon) + \mathcal{N}(C \cup W(C) \cup W^2(C) \cup W^3(C) \cup \ldots \cup W^i(C), \epsilon)) \\
&= \Theta_R(\epsilon^{-D} + \sum_{t=0}^{i} \epsilon^{-D}) \qquad \text{(from the claim (5))} \\
&= \Theta_R(\epsilon^{-D} + \epsilon^{-D} \log \epsilon^{-1}) \qquad \text{(since } i = c_2 \log \epsilon^{-1}) \\
&= \Theta_R(\epsilon^{-D} \log \epsilon^{-1}).
\end{aligned}$$

This completes the proof of the theorem.

LEMMA 3. *Consider the statement of Theorem 4. Let*

$$\mathcal{N}(C, \epsilon) = \Theta_R(\epsilon^{-D_1} \log^p \epsilon^{-1}).$$

Let $D_1 = D_2 = D$. Then

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D} \log^{p+1} \epsilon^{-1}).$$

PROOF. The proof is same as that of Theorem 4 till the point where we make the claim (5). Let $s_L = \min\{s_i\}$ and $s_U = \max\{s_i\}$.

In the following all logarithms are base b, where b is chosen so that $b > 1/s_L$.

Now we claim that for each $i \geq 0$,

$$k_1 \epsilon^{-D} \log^p(\epsilon/s_L^i)^{-1} < \mathcal{N}(W^i(C), \epsilon) < k_2 \epsilon^{-D} \log^p(\epsilon/s_U^i)^{-1}. \tag{6}$$

The basis of the proof by induction remains same. For the induction step, note that

$$k_1 \left(\frac{\epsilon}{s_j}\right)^{-D} \log^p(\epsilon/(s_L^i s_j))^{-1} \leq \mathcal{N}\left(W^i(C), \frac{\epsilon}{s_j}\right) \leq k_2 \left(\frac{\epsilon}{s_j}\right)^{-D} \log^p(\epsilon/(s_U^i s_j))^{-1},$$

and therefore

$$k_1 \left(\frac{\epsilon}{s_j}\right)^{-D} \log^p(\epsilon/s_L^{i+1})^{-1} \leq \mathcal{N}\left(W^i(C), \frac{\epsilon}{s_j}\right) \leq k_2 \left(\frac{\epsilon}{s_j}\right)^{-D} \log^p(\epsilon/s_U^{i+1})^{-1},$$

from which the claim (6) follows for $i + 1$. An interesting property of $\log^p \epsilon^{-1}$ with base b is that for any a such that $1/b < a < 1$,

$$\sum_{j=0}^{\log \epsilon^{-1}} \log^p(\epsilon/a^j)^{-1} = \Theta_R(\log^{p+1} \epsilon^{-1}).$$

In the above, we assume without loss of generality that $\log \epsilon^{-1}$ is an integer (the proof can be otherwise modified by considering $\lfloor \log \epsilon^{-1} \rfloor$ or $\lceil \log \epsilon^{-1} \rceil$). To see why the above is true, note that each term on left-hand side is of the form

$$\log^p(\epsilon/a^j)^{-1} = (\log \epsilon^{-1} + j \log a)^p.$$

But since $1/b < a < 1$, therefore $-1 < \log a < 0$. Thus $0 < 1 + \log a < 1$. Thus since j takes values from 0 to $\log \epsilon^{-1}$, for all j,

$$(1 + \log a)^p \log^p \epsilon^{-1} \leq (\log \epsilon^{-1} + j \log a)^p \leq \log^p \epsilon^{-1}.$$

Thus there exist positive real constants $c_1 = (1 + \log a)^p$ and $c_2 = 1$ such that

$$c_1 \log^{p+1} \epsilon^{-1} \leq \sum_{j=0}^{\log \epsilon^{-1}} \log^p(\epsilon/a^j)^{-1} \leq c_2 \log^{p+1} \epsilon^{-1}.$$

Since $s_L, s_U > 1/b$, the above holds for $a = s_L$ and $a = s_U$ and from (6)

$$\mathcal{N}(C \cup W(C) \cup \ldots \cup W^{-\log \epsilon}(C), \epsilon) = \Theta_R(\epsilon^{-D} \log^{p+1} \epsilon^{-1}).$$

Therefore, continuing the proof of Theorem 4 and choosing $i = c \log \epsilon^{-1}$ to be such that ϵ-balls covering $C, W(C), \ldots, W^i(C)$ cover also $W^j(C)$ for all $j > i$,

$$\mathcal{N}(C \cup W(C) \cup W^2(C) \cup \ldots, \epsilon) = \mathcal{N}(C \cup W(C) \cup \ldots \cup W^i(C), \epsilon)$$
$$= \Theta_R(\epsilon^{-D} \log^{p+1} \epsilon^{-1}).$$

From this and (4) the lemma follows. Note that in Θ_R notation, one can again have the natural logarithm.

Now one can easily generalize the above results when we have an MRFS with two strongly connected components which may have more than one nodes.

THEOREM 5 *Let $M = (V, \phi, W, G)$ be an MRFS having attractor O. Let $SC(G) = \{H_1, H_2\}$. Let there be no nodes in G other than those in H_1 and H_2. Let there be a single arc from H_1 to H_2, labeled with similitude w. Let O_i be the attractor of the MRFS $M_i = (V_i, \phi, H_i)$ for $i = 1, 2$. Let the Hausdorff-Besicovich dimensions of O_1 and O_2 be D_1 and D_2 respectively. Then, for all $\epsilon \in [0, 1]$, if $D_1 \neq D_2$,*

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-\max\{D_1, D_2\}}),$$

otherwise if $D_1 = D_2 = D$,

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D} \log \epsilon^{-1})$$

PROOF. The proof for the case when $D_1 \neq D_2$ follows directly from the Mauldin-Williams Theorem. We prove the case when $D_1 = D_2 = D$. The proof is a generalization of Theorem 4 and results in [3, 15].

Let the arc from H_1 to H_2 be from the node E (in V_1) to node F (in V_2). Let E' be the limiting value of E (the image defined by E). Let $C = w(E')$. Now Theorem 4 can be generalized as follows: treat H_1 as a condensation set E' connected to F with an edge labeled w.

Formally, let $|V_2| = n$. Consider the Hausdorff metric on tuples of sets as

$$H((A_1, A_2, \ldots, A_n), (B_1, B_2, \ldots, B_n)) = \max\{h(A_i, B_i) | i = 1, 2, \ldots, n\}.$$

Since MRFS are loop contractive, therefore there exists $k \geq 1$ such that $W^{(k)}$, the k-fold composition of the mapping W, is contractive. Thus we can continue the proof of Theorem 4 in a parallel fashion and using the results on dimension of attractors of strongly connected MRFS in [3, 15]. Here we work with tuples in $(\mathcal{H}(X))^n$ and the claim (5) is made for each component of these tuples.

Let $(x_1, x_2, \ldots, x_n)$ be a strictly positive eigenvector of S_β matrix of M_2 corresponding to the eigenvalue 1, according to the Perron-Frobenius Theorem [10] in

Theorem 1. Formally, if V_i^t denotes the i-th component of tuple after t iterations of the Deterministic Algorithm, then we claim that there exist positive constants C_1 and C_2 such that

$$C_1 \epsilon^{-D} x_i \leq \mathcal{N}(V_i^t, \epsilon) \leq C_2 \epsilon^{-D} x_i. \tag{7}$$

For the induction hypothesis, assume that (7) holds for some $t \geq 0$. For the induction step, consider $t + 1$. Now for each $i = 1, 2, \ldots, n$,

$$V_i^{t+1} = \bigcup_{c_{i,j}=1} w_{i,j}(V_j^t).$$

Since M is nonoverlapping,

$$\mathcal{N}(V_i^{t+1}, \epsilon) = \sum_{c_{i,j}=1} \mathcal{N}\left(V_j^t, \frac{\epsilon}{s_{i,j}}\right).$$

From the induction hypothesis,

$$\mathcal{N}(V_i^{t+1}, \epsilon) \leq \sum_{c_{i,j}=1} C_2 \left(\frac{\epsilon}{s_{i,j}}\right)^{-D} x_j$$

$$= C_2 \epsilon^{-D} \sum_{c_{i,j}=1} s_{i,j}^D x_j$$

$$= C_2 \epsilon^{-D} x_i \qquad \text{(since } (x_1, x_2, \ldots, x_n) \text{ is an eigenvector).}$$

Similarly, we show that

$$C_1 \epsilon^{-D} x_i \leq \mathcal{N}(V_i^{t+1}, \epsilon).$$

The rest of the proof follows along lines parallel to the proof of Theorem 4.

LEMMA 4. *Consider the statement of Theorem 5. Let*

$$\mathcal{N}(O_1, \epsilon) = \Theta_R(\epsilon^{-D_1} \log^p \epsilon^{-1}).$$

Let $D_1 = D_2 = D$. Then

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D} \log^{p+1} \epsilon^{-1}).$$

PROOF. The proof is a straightforward generalization of the proof of Lemma 3.

THEOREM 6 *Let $M = (V, \phi, W, G)$ be an MRFS. Let the attractor O of M have Hausdorff-Besicovich dimension equal to D. Let $H_1 H_2 \ldots H_{p+1}$ be a sequence of maximal length p, such that $H \in SC(G)$, $\alpha_{H_i} = D$ for each $i \in \{1, 2, \ldots, p + 1\}$, and there is a path from H_j to H_{j+1} for $j \in \{1, 2, \ldots, p\}$ in G (not passing through any $H \in SC(G)$). Then*

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D} \log^p \epsilon^{-1}).$$

PROOF. The proof consists of one application of Theorem 5 followed by repetitive applications of Lemma 4. First consider the path $H_1 H_2$, and then $H_1 H_2 H_3$ and so forth.

6. Modeling algorithms by MRFS

A group $A = \{A_1, A_2, \ldots, A_n\}$ of mutually recursive algorithms can be modelled by or viewed as a condensation MRFS $M = (V, C, W, G)$, where

$$V = \{V_1, V_2, \ldots, V_n\},$$

the set V_i represents the algorithm A_i, $i = 1, 2, \ldots, n$, and

$$C = \{C_1, C_2, \ldots, C_n\},$$

where the condensation set C_i represents the additional computation done by the algorithm A_i. C_i is a condensation set with Hausdorff-Besicovich dimension equal to D_i if the algorithm A_i does additional computation of $\Theta(n^{D_i})$ steps. If $D_i = 0$ then we ignore C_i.

The underlying labeled digraph G of M represents the interrelationships between algorithms and the additional computation performed by them.

Consider each algorithm $A_i \in A$. Suppose the algorithm A_i calls recursively the algorithms $A_{j_1}, A_{j_2}, \ldots, A_{j_r}$, and at these recursive calls the input size is contracted by a factor $s_1, s_2, \ldots, s_r$, respectively. Then the component V_i representing the algorithm A_i is mutually recursively defined as

$$V_i = w_{i_1}(V_{j_1}) \cup w_{i_2}(V_{j_2}) \cup \ldots \cup w_{i_r}(V_{j_r}) \cup C_i,$$

where w_{i_k} is a similitude with contractivity factor equal to s_k, $k = 1, 2, \ldots, r$.

If one needs to generate the actual fractal which geometrically captures the working of the algorithm, then one needs to choose the affine transformations and the condensation images. These have to be chosen so that the condensation MRFS M is nonoverlapping. However if one needs to only determine the time complexity of A, then one needs only the contractivities of the affine transformations and the Hausdorff-Besicovich dimensions of the condensation images.

EXAMPLE 3 Refer to Example 1 and Figure 4. This group of recursive algorithms can be modelled by an MRFS $\mathcal{M}$ with 5 components (vertices in the underlying digraph) M, S1, S2, S3 and S4 representing the 5 routines and 3 condensation sets C_1, C_2 and

C_3 representing the additional computation performed by subroutines Sub2, Sub4 and Sub3, respectively. The MRFS is specified by:

$$M = w_1(M) \cup w_2(M) \cup w_3(M) \cup w_4(M) \cup u(S2) \cup v(S3),$$
$$S1 = w_5(S1) \cup w_6(S1) \cup w_7(S2) \cup w_8(S2),$$
$$S2 = w_9(S1) \cup w_{10}(S2) \cup w_{11}(S2) \cup C_1,$$
$$S3 = w_{12}(S3) \cup w_{13}(S3) \cup w_{14}(S3) \cup w_{15}(S4) \cup C_3,$$
$$S4 = w_{16}(S4) \cup w_{17}(S4) \cup w_{18}(S4) \cup w_{19}(S4) \cup C_2.$$

The transformations $w_1, w_2, \ldots, w_{19}$ have contractivity factors equal to $\frac{1}{2}$, and u and v have equal to 1 and 2 respectively. The fractal dimensions of the condensation sets C_1, C_2 and C_3 are 2, 2 and 1 respectively.

Just to facilitate our discussion and without any loss of generality, we will assume that the attractor of M is a subset of the unit box $U = [0, 1]^k$ of the underlying k-dimensional Euclidean space. Also one can choose transformations which just scale and translate, so they map a box into a box. In the Deterministic Algorithm, let the initial compact subsets of M be all the unit box U.

Now a recursive call during the execution of A will correspond to the image of a box during the execution of the Deterministic Algorithm on M. The size of the input in this call will correspond to the size of the side of the box. A chain of recursive calls will correspond to a sequence of affine transformations of M which map U into a box.

Let the initial input size be n. Now if there is a chain of recursive calls $A_1, A_2, \ldots, A_p$ then at the end of this chain, the size of the input will be scaled down to

$$n \times s_1 \times s_2 \times \ldots \times s_p,$$

where s_i is the input scaling factor of the call A_i. Therefore, after p iterations of the Deterministic Algorithm, the size of the side of the corresponding box will be

$$1 \times s_1 \times s_2 \times \ldots \times s_p,$$

since the size of side of the starting unit box U is 1. Thus if $\text{Calls}(A, m)$ denotes the number of chains of recursive calls which scale down the input to size m, and if $\text{Boxes}(M, \epsilon)$ denotes the number of boxes with side length ϵ, then

$$\text{Boxes}(M, \epsilon) = \text{Calls}(A, \epsilon n). \tag{8}$$

EXAMPLE 4 Consider the recursive algorithm A is shown in Figure 1.

Now A can be modelled by a single component MRFS (IFS) M. We select 2-D Euclidean space and three of the four quadrant transformations, each with contractivity

0.5, which map the unit square $U = [0, 1]^2$ into the lower-left, lower-right and upper-left quadrants, respectively.

In Figure 5, we show the first 3 steps of the Deterministic Algorithm on M and the attractor of the Sierpinski Triangle which corresponds to the asymptotic case. The initial starting image for the Deterministic Algorithm is chosen to be the unit square U. Each box in Figure 5 corresponds to some chain of calls made during the execution of A.

Note that the modelling of algorithms by MRFS is not a geometric interpretation of algorithms as the only relevant part of the geometry is the contractivity factors. The deterministic algorithm simply takes an input string, which can be thought of as real or integer, and divides it by a factor after each call. Thus the embedding in k-dimensional space seems to serve merely as a method of producing a fractal image.
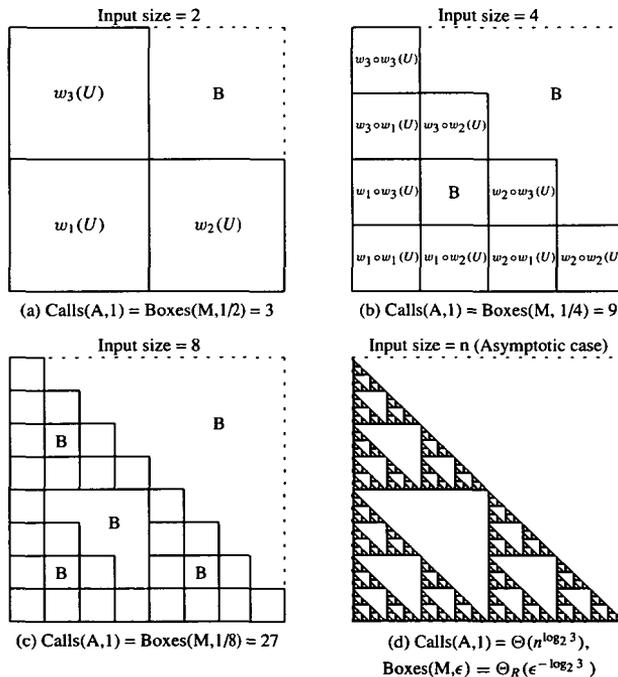

FIGURE 5. Illustration of correspondence of chains of recursive calls in the execution of a recursive algorithm and of boxes in the execution of the modelling recursive function system. U is the unit square $[0, 1]^2$ and is shown by the dashed lines. B denotes a blank region.

7. The main result

Since a group of recursive algorithms can be viewed as a condensation MRFS, so we can naturally generalize the concepts associated with the latter to the former. Let A be a group of mutually recursive algorithms with underlying digraph G.

We make the assumption that there is at most one arc from one vertex to another in G. This does not result in any loss of generality, as stated in Theorem 3.

From the Mauldin-Williams Theorem, the Hausdorff-Besicovich dimension of the attractor of A (viewed as a condensation MRFS) is $\alpha = \max\{\alpha_H | H \in SC(G)\}$. Consider all those strongly connected components of G with Hausdorff-Besicovich dimension equal to α,

$$SCMAX(G) = \{H \in SC(G) | \alpha_H = \alpha\}.$$

Now from G we construct a reduced digraph G' by collapsing elements of SCMAX (G) into nodes. For each H in SCMAX (G) we create a node v_H in G'. If there is path P in G from H to K where $H, K \in$ SCMAX (G) and H and K are distinct, such that the path P does not pass through any other component in SCMAX (G), then we place an arc from v_H to v_K.

Then G' so obtained is a directed acyclic graph (DAG) and is called the *order structure DAG* of A.

In the following theorem, the function Φ is the same as defined in Theorem 1. That is, $\Phi(D)$ is the modulus of the largest eigenvalue of the matrix obtained by raising each element of the matrix S to the power o D, where $S_{i,j}$ is the contractivity factor of recursive call of the algorithm j by algorithm i.

THEOREM 7 *Suppose A is a group of mutually recursive algorithms such that the underlying graph is strongly connected. Let $T(n)$ be the time complexity of A on input of size n. Then*

$$T(n) = \Theta(n^D),$$

where $D = \alpha_A$ is the nonnegative real number for which $\Phi(D) = 1$.

PROOF. Let A be viewed as a condensation MRFS M. Let O be the attractor. Since the computation is done at the trivial-case when $n = 1$, therefore from (8)

$$T(n) = \text{Calls}(A, 1) = \text{Boxes}(M, 1/n).$$

Let $\epsilon = 1/n$. From the Box-Counting Theorem for the computation of fractal dimension in [2],

$$\text{Boxes}(M, \epsilon) = \Theta_R(\mathcal{N}(O, \epsilon)).$$

Therefore,

$$c_1 \mathcal{N}(O, 1/n) \leq T(n) \leq c_2 \mathcal{N}(O, 1/n),$$

for some positive constants c_1 and c_2 and sufficiently large n.

Since A is strongly connected, therefore from the Mauldin-Williams Theorem

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D}),$$

where $D = \alpha_A$ is the nonnegative real number for which $\Phi(D) = 1$. Therefore

$$T(n) = \Theta(n^D).$$

Furthermore from Lemma 2, the Hausdorff D-dimensional measure of the attractor of A is finite and is the constant factor by which the above inequality is asymptotically bounded from above, and thus is the constant of proportionality by which two algorithms with same value of D can be compared.

THEOREM 8 *Let A be a group of mutually recursive algorithms with underlying digraph G. Let $T(n)$ be the time complexity of A on input of size n. Then*

$$T(n) = \Theta(n^D \log^p n),$$

where

$$D = \max\{\alpha_H | H \in \mathrm{SC}\,(G)\}$$

and p is the length of the longest path in the order structure DAG of M.

PROOF. The case $p = 0$ is Theorem 7. Let $p > 1$. Consider the proof of Theorem 7. Since the computation is done at the trivial-case when $n = 1$, and the additional computations are also represented by the condensation sets, therefore, from Theorem 6,

$$\mathcal{N}(O, \epsilon) = \Theta_R(\epsilon^{-D} \log^p \epsilon^{-1}),$$

where

$$D = \max\{\alpha_H | H \in \mathrm{SC}\,(G)\}.$$

Therefore, substituting $\epsilon^{-1} = n$,
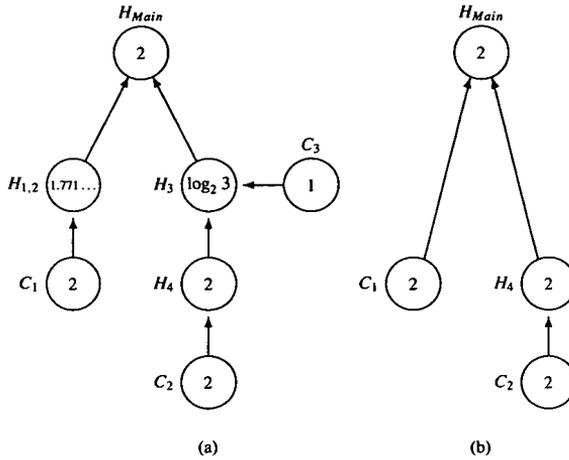
$$T(n) = \Theta(n^D log^p n).$$

FIGURE 6. (a) Fractal dimension of strongly connected components of MRFS modelling algorithms in Figure 4 (b) Order structure DAG with length of the longest path equal to 2.

EXAMPLE 5 Refer to Examples 1 and 3. How can we determine the time complexity of the algorithms? We need to solve the following set of recurrence relations:

$$T_M(n) = 4T_M(n/2) + T_{S2}(n) + T_{S3}(2n),$$
$$T_{S1}(n) = 2T_{S1}(n/2) + 2T_{S2}(n/2),$$
$$T_{S2}(n) = T_{S1}(n/2) + 2T_{S2}(n/2) + \Theta(n^2),$$
$$T_{S3}(n) = 3T_{S3}(n/2) + T_{S4}(n/2) + \Theta(n),$$
$$T_{S4}(n) = 4T_{S4}(n/2) + \Theta(n^2).$$

The conventional methods based upon discrete analysis, such as the Master Theorem in [6], seem to be inadequate. However using the above theorem one can easily solve the recurrence relations. In Figure 6(a), we show the strongly connected components of the MRFS M in Example 3. In total there are 7 such components. Each of these components is an MRFS by itself and defines an image. Using Corollaries 1 and 2, one quickly computes the fractal dimension of these images. For example, the fractal dimension of the image defined by the component $H_{1,2}$ is $\log_2(2+\sqrt{2})$, where $2+\sqrt{2}$ is obtained as the largest eigenvalue of the connection matrix

$$C = \begin{bmatrix} 2 & 1 \\ 2 & 2 \end{bmatrix}.$$

The number inside a circle in Figure 6(a) is the fractal dimension of the image defined by the corresponding component. Since the maximum of the fractal dimensions is 2,

we construct the order structure DAG as shown in Figure 6(b) by keeping only the components with fractal dimension equal to 2. The length of the longest path in the DAG is 2. Therefore, substituting $D = 2$ and $p = 2$ in the theorem, we obtain the solution

$$T_M(n) = \Theta(n^2 \log^2 n).$$

8. Conclusions

This paper made explicit a relationship between fractal geometry and divide-and-conquer recurrences which leads to a generalization of known results on the latter.

1. We assumed that an algorithm can perform an additional computation (which for example may involve the linear time taken to read the input) taking $\Theta(n^D)$ steps where D is nonnegative real number. Theorem 8 can be easily extended to the case when this additional computation has time complexity of the form $\Theta(n^D f(n))$ where $f(n)$ is a power of the logarithmic function, that is, $\log^k n$. Just use Lemma 4.
2. A special case, when the MRFS is a condensation IFS, of Theorem 8 is the Master Theorem as stated in [6]. Note how easy a corollary is the latter of the former and compare this with the lengthy proof of the latter in [6].
3. An interesting correlation can be made between overlapping MRFS and certain parallel algorithms, and between MRFS defining grey (colour) fractals and certain randomized algorithms by associating probabilities to the transformations.

In conclusion, it is pleasing to note that these two different fields, algorithms and fractal geometry, intersect in a useful way providing us with new insights and leading to new results. Note that though these results can be *also* derived in the discrete domain, however then one would spend much more effort by not utilizing already known results from the continuous domain and would miss this interesting connection.

References

[1] A. V. Aho, J. E. Hopcroft and J. D. Ullman, *The design and analysis of computer algorithms* (Addison-Wesley, 1974).
[2] M. F. Barnsley, *Fractals everywhere* (Academic Press, 1988).
[3] M. F. Barnsley, J. H. Elton and D. P. Hardin, "Recurrent iterated function systems", *Constructive Approximation* 5 (1989) 3–31.
[4] M. F. Barnsley, R. L. Devaney, B. B. Mandelbrot, H.-O. Peitgen, De Saupe and R. F. Voss, *Science of fractal images* (Springer-Verlag, 1988).

[5] J. L. Bentley, D. Haken and J. B. Saxe, "A general method for solving divide-and-conquer recurrences", *SIGACT News* **12** (1980) 36–44.

[6] T. H. Cormen, C. E. Leiserson and R. L. Rivest, *Introduction to algorithms* (MIT Press, 1990).

[7] K. Culik II and S. Dube, "Affine automata and related techniques for generation of complex images", *Theoretical Computer Science* **116** (1993) 373–398.

[8] K. Culik II and S. Dube, "Rational and affine expressions for image synthesis", *Discrete Applied Mathematics* **41** (1993) 85–120.

[9] K. Culik II and S. Dube, "Balancing order and chaos in image generation", *Computer and Graphics* **17** (4) (1993) 465–486.

[10] F. R. Gantmacher, *Applications of the theory of matrices* (Interscience Publishers, 1959).

[11] R. L. Graham, D. E. Knuth and O. Patashnik, *Concrete mathematics* (Addison-Wesley, 1989).

[12] D. H. Greene and D. E. Knuth, *Mathematics for the analysis of algorithms* (Birkhäser, Boston, 1982).

[13] J. Hutchinson, "Fractals and self-similarity", *Indiana University J. of Mathematics* **30** (1981) 713–747.

[14] B. Mandelbrot, *The fractal geometry of nature* (W. H. Freeman and Co., San Francisco, 1982).

[15] R. D. Mauldin and S. C. Williams, "Hausdorff dimension in graph directed constructions", *Trans. AMS* **309** (1988) 811–829.

[16] P. W. Purdom, Jr. and C. A. Brown, *The analysis of algorithms* (Holt, Rinehart and Winston, 1985).

[17] L. Staiger, "Quadtrees and the Hausdorff dimension of pictures", in *Workshop on geometrical problems of image processing (GEOBILD'89)*, Math. Research No. 51, (Akademie-Verlag, Berlin, 1989), 173–178.