J. Phys. A: Math. Gen. **39** (2006) 1609–1618 doi:10.1088/0305-4470/39/7/006

The coastline and lake shores on a fractal island

Peter Blaudeck[1], **Steffen Seeger**[1], **Christian Schulzky**[1], **Karl Heinz Hoffmann**[1], **Tapati Dutta**[2] **and Sujata Tarafdar**[3]

[1] Institut für Physik, Technische Universität, 09107 Chemnitz, Germany
[2] Physics Department, St Xavier's College, Kolkata 700 016, India
[3] Condensed Matter Physics Research Centre, Jadavpur University, Kolkata 700 032, India

Received 26 September 2005
Published 1 February 2006
Online at stacks.iop.org/JPhysA/39/1609

Abstract
We compute the fractal dimensions of the 'hulls' or external boundary and the boundaries of the internal cavities in several deterministic as well as random fractal structures. Our conclusion is that the two fractal dimensions are in fact identical. The deterministic fractals we study are Sierpinski carpets (SC) in a two-dimensional space and the random fractals are percolation clusters at criticality. As an intermediate case, we present results on some randomized SC. In the random structures, statistics of the area and perimeters of all internal cavities or holes are taken and the fractal dimension of the hull borderline is computed. Two different definitions of the borderline are used, considering nearest neighbours as well as nearest *and* second nearest neighbours as connected. The conclusion is valid for both cases.

PACS numbers: 61.43.Hv, 89.75.Da, 64.60.Ak

1. Introduction

Natural growth processes very often lead to self-similar systems with fractal dimensions [1, 2]. Examples are sponges, corals, vacuum-deposited surfaces and so on. Many of these objects have cavities or holes inside, developed as a result of the same natural process, which generate the structure. The holes also show fractal characteristics. For Euclidean objects with embedded holes, we expect the dimension of the outer surface to be the same as the dimension of the internal surface bounding the hole. It is quite clear that an object in a three-dimensional space has a two-dimensional external surface, and any hole in it is also bounded by a two-dimensional surface. An object embedded in a two-dimensional space, as well as any hole in it will have a linear boundary. The corresponding situation in fractals is not so obvious. Is it necessary for a porous material to have an external and internal interface with the same fractal dimension? Is it at least usual? The present work gives a tentative answer to this question.

It is well known that, for mass fractals, the *total* interface including external as well as all internal boundaries scales exactly as the total mass [1, 3]. The external perimeter alone

0305-4470/06/071609+10$30.00

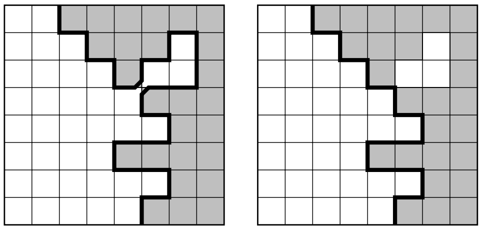

Figure 1. Definitions of the borderline, left: natural borderline, right: nearest-neighbour borderline.

scales with a different fractal dimension, usually referred to as the 'hull dimension' [3, 4]. Franz *et al* [5] showed that for the Sierpinski carpet (SC) holes are characterized by a fractal dimension d_p. We demonstrate first that for the Sierpinski carpet family with internal holes, the fractal dimension of the hole boundaries d_p is indeed equal to the fractal dimension of the external boundary. This does not imply, of course, that this is a general result, especially it may not hold for random fractals, and hence for natural systems. Then we turn to the case of a typical and much-studied random fractal—the percolation cluster at criticality [6] which is still a subject of intensive research [7–9], but to our knowledge statistics of the holes or cavities has not received any attention. The percolation cluster has to be studied by numerical methods and exact answers cannot be obtained as in the case of the Sierpinski carpets. It appears, however, that within the limits of error the fractal dimension of the hole boundaries is equal to the well-known 'hull dimension', i.e. the dimension of the external perimeter. So, our conjecture is that this identity of the internal and external perimeter dimensions is a general result. As an intermediate case, we also study the randomized Sierpinski carpets, where some randomness is introduced into the pattern. Such systems have been studied previously under the name 'regular random fractals' [10, 11, 13].

2. Borderlines

Let us recall the standard percolation problem [6] on a square lattice. Unit squares are randomly occupied on a square lattice with probability w. Usually one is interested in the condition for percolation, i.e. the value of $w = w_c$, when a spanning cluster appears, connecting opposite boundaries of the system. If the system is realized on a computer, each unit square is a pixel at the highest resolution. At present, we are concerned with a system where $w = w_c$, i.e. at criticality, when the largest spanning cluster is known to be a fractal [3]. Let us look at the spanning cluster. The common definition of a two-dimensional site percolation cluster is that two occupied pixels, which only touch each other at one corner, are *not* connected. Therefore, as a logical consequence, two vacant pixels, which touch each other at one corner, *are* connected and belong either both to the outer space or both to the same hole. In other words, a vacant pixel is connected with vacant pixels being nearest or next nearest neighbours. This is the way to define a 'natural' borderline. However, for some physical applications, for example transport through holes, there is another sensible definition of a so-called 'nearest neighbour'-borderline. In that case it is suggested that particles cannot pass the (theoretically infinitesimal) small gap between two occupied pixels touching at one corner. Within this definition, two vacant pixels are only assumed to be connected if they are nearest neighbours, see figure 1.

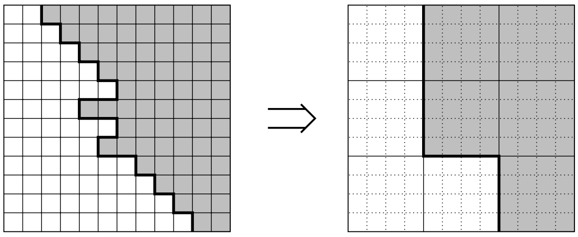

Figure 2. Increasing the ruler for measuring the length of a borderline. The property of each new 'superpixel' (vacant or not) is determined by the majority of its 4 × 4 primary pixels.

The area of a hole a is the number of its pixels. The perimeter is the number of edges, which form the borderline. In our sample of hundreds of clusters we have an excellent statistics provided by millions of holes. This means, the dimension d of hole perimeters p can easily be determined by applying the regression formula

$$p \sim a^{\frac{d}{2}} \tag{1}$$

as realized in sections 4.2 and 5.

3. Hull dimension

In the case of the hull, we have to extract the dimension from much less information, i.e. (a) only one hull per cluster with (b) nearly the same extent for all clusters. Therefore, to calculate the fractal dimension of these borderlines, just in the same way as for geographical islands: increase, step-by-step, the length l of a ruler, measure repeatedly the length of the borderline L, and fit the results to the function $L \sim l^{(1-d)}$. To implement this procedure for our structure, we overlay the system by a grid with a side length larger than the pixel size. This length is the new l. The grid is our new structure. Now we look at the pixels using some kind of matt screen as illustrated in figure 2. For example, if the grid length is increased by a factor 4, the 16 pixels of each grid cell are transformed to one new but larger 'superpixel'. The property of that new pixel is determined by the majority of the former small pixels. If there is no majority that property is specified by 'flipping a coin'.

4. Sierpinski carpets and the hole-perimeter dimension

Now we turn to a family of deterministic fractals, the well-known Sierpinski carpets (SC). Many different aspects of these fractals have been studied [14–16]. SC are simple to construct in any dimension and properties such as the fractal dimension and connectivity can be tuned to desired values. This makes the SC a convenient choice for a prototype deterministic fractal which may serve as a model for natural fractals, e.g. porous media. One can make the resemblance closer by randomizing the SC [10, 11, 13]. This makes the system stochastic to a certain extent, while still tractable and having a tunable fractal dimension. Sierpinski carpets are constructed in the following way: one starts with a unit square, divides it into $n \times n$ congruent smaller subsquares and removes $(n^2 - m)$ of them. That gives an $n \times n$ pattern called the generator of the Sierpinski carpet. This construction procedure is repeated with all the remaining subsquares *ad infinitum*. The resulting object is a fractal of fractal dimension

$d_f = \log(m)/\log(n)$, called a Sierpinski carpet. This can be generalized to three dimensions to a Menger sponge [1], but at present we remain confined to two-dimensional carpets.

Sierpinski carpets can be finitely or infinitely ramified like other deterministic fractals [17]. The ramification may be different in the vertical and horizontal directions, and is determined by how many occupied squares in the generator connect with the neighbouring block, if a replica of the generator is placed beside it. These are the 'connection points' of the generator. If the number of connection points is 1 in the vertical as well as horizontal direction, the fractal is finitely ramified. If it is more than 1 in either direction, ramification is infinite. Further details are discussed in [5]. If the resulting carpet has a connected path from one end to the other, after the required number of iterations, we say it is 'percolating'. An example is shown in figure 3. The generator of that carpet was a fixed 4×4 pattern, with 12 occupied positions. All parts of the carpet which are not connected with the percolating cluster have been removed.

4.1. Deterministic Sierpinski carpets

It has already been shown that the pores within Sierpinski carpets have a definite fractal dimension between 1 and 2 which can be exactly determined from the generator [5]. Below we show that the external boundary has the same fractal dimension as the holes.

Before we derive a formula for the total perimeter of the ith iteration of the carpet iterator, let us introduce the notation used. Let I be an $n \times n$ iterator $(i = 1)$ of the carpet of interest and m be the number of tiles in the iterator. We denote by r_H and r_V the horizontal and vertical ramification of the carpet, respectively, that is the number of horizontally and vertically connecting surface tiles. s_H and s_V denote the number of horizontal and vertical inner surfaces, respectively. Then we have for the number of surfaces of all tiles in the ith iteration

$$s_i^T = m^i s_0^T \tag{2}$$

with s_0^T being the number of surfaces of one single tile ($s_0^T = 4$ for quadratic tiles).

Next we count all inner surfaces of the ith iteration. We start with a single tile which, of course, has no inner surface and thus $s_0^I = 0$. Then the ith iteration has

$$s_i^I = m s_{i-1}^I + r_H^{i-1} s_H + r_V^{i-1} s_V \tag{3}$$

inner surfaces because we have m replicas of the iterator of stage $(i - 1)$ and (ramification $\times$ inner surfaces) new inner surfaces in vertical and horizontal direction because of the new connections appearing in the next stage. From the hole counting polynomial (see [5]), we can calculate the total perimeter of all inner loops s_i^L.

Thus we have for the outer perimeter s_i of the ith carpet iteration

$$s_i = s_i^T - 2s_i^I - s_i^L \tag{4}$$

because the outer perimeter is all that is left when we count the surfaces of all tiles, subtract twice the number of inner surfaces and all inner loop surfaces. If we examine the scaling behaviour of the carpet perimeter, we find that it scales with the pore dimension d_p. For all iterators that generate pores, we either have a pore in the iterator itself or it is generated from a pattern of tiles after the iteration. In the latter case, we can easily see that the pore surface is composed of exactly the surface of the iterator minus the surfaces of the ramification points (see figure 4). Thus the surface becomes an inner loop after the first iteration and thus we could expect its surface to scale just as an inner loop. For inner loops already present in the iterator, we would expect the same except that we will be able to map the iterator surface several times onto the loop surface. Interestingly, the surfaces of the ramification points appear only once.

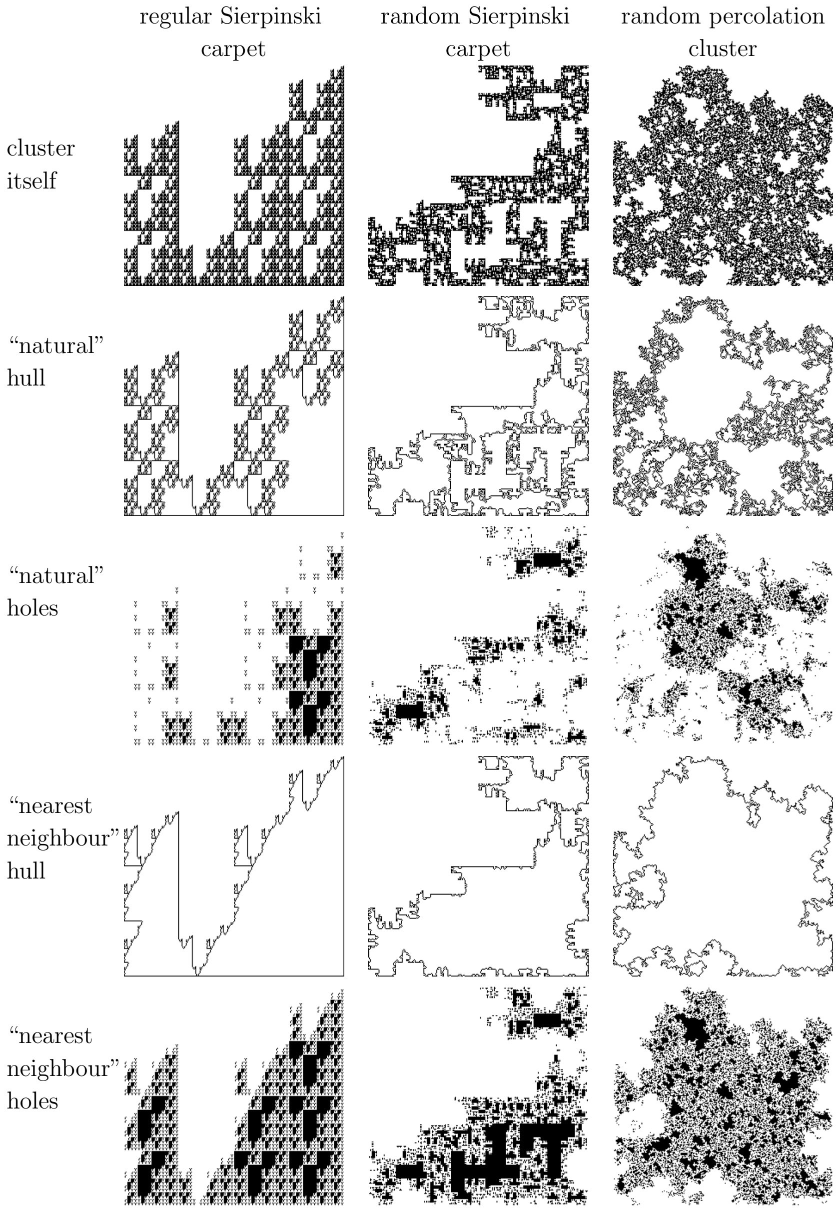

Figure 3. Examples for: a regular Sierpinski carpet (left), a random Sierpinski carpet (middle), and a random percolation cluster (right). Below each cluster: the hulls and the corresponding holes for both types of borderline, 'natural' and 'nearest neighbour'.

4.2. Random Sierpinski carpets

The construction of random Sierpinski carpets is similar to that of regular ones. The generator is divided as before into $n \times n$ congruent squares, m are occupied and $(n^2 - m)$ being vacant. In the next stage, each remaining occupied subsquare is divided into $n \times n$ equal subsquares, but the assignment—whether it is occupied or vacant—is based on different randomly chosen

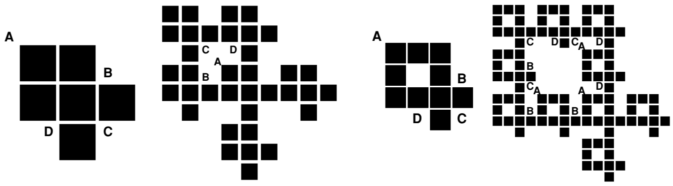

Figure 4. Two examples for the first iteration.

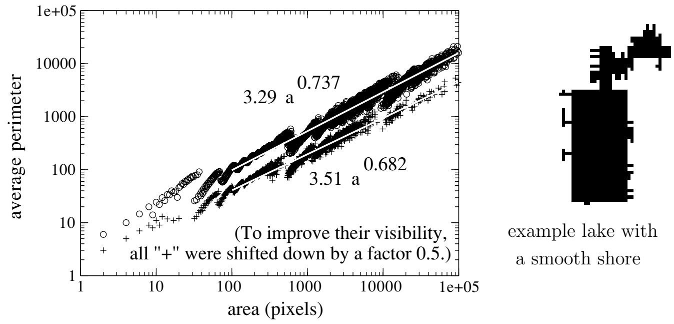

Figure 5. Average perimeter for each area for random Sierpinski carpets 1024 × 1024, together with the results of regressions (white lines), circles: natural borderlines, crosses: nearest-neighbour-borderlines. Exponents of the power-laws shown are half the fractal dimension according to (1)

generators each time. This procedure is repeated *ad infinitum* or up to a desired iteration depth. An example is shown in figure 3. It is to be noted that deterministic SC can be constructed so as to have no isolated occupied site, i.e. with all occupied sites belonging to one spanning cluster, but this is not possible for randomized SC. So arguments valid for deterministic SC in section 4.1 are not directly applicable to randomized SC. In figure 5, we present the results for a set of 500 equally dense and random Sierpinski carpets, 1024 × 1024 each. As generators we have randomly used a set of 16 patterns, each 4 × 4 with 12 occupied positions. Only if the resulting cluster connects two opposite sides of the 1024 × 1024 square, the connecting part of the cluster will be further analysed. The discontinuities of the functions $p(a)$ are due to qualitatively new lakes with particularly smooth perimeters. These lakes contain large and coarse patterns from early steps of carpet generation, as the big rectangle in the example lake on the right of figure 5. For better reliability, the regressions following (1) have been performed in all cases only for lakes containing 100 or more pixels, neglecting millions of very small lakes which would falsify the statistics due to their angular non-fractal perimeters.

The randomized SC are in between deterministic and random fractals. We consider next a truly stochastic fractal—the percolation cluster at criticality.

5. Random percolation clusters

5.1. Generation and analysis

To create a connected cluster at the percolation threshold, at first the sites of a square lattice are occupied randomly with a probability $w = 0.592\,746$, which is the site percolation threshold on a square lattice [6]. Two occupied sites are connected, if they touch each other by one of the edges. Then, if there is a continuous connection from one end of the system to the opposite

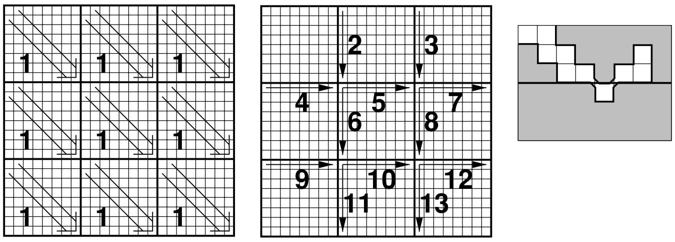

Figure 6. The modified Hoshen–Kopelman analysis, simplified for a 3 × 3 partitioning. Left: first step, complete, parallel, and independent analysis of smaller subsquares, right: sequential renumbering by analysing the borderlines of the subsquares (example detail: two holes of an upper subsquare are newly linked by a hole of a lower subsquare).

side, a spanning cluster exists and all other occupied sides are eliminated, so that only this cluster with its holes remains. An example is shown in figure 3.

To analyse the large amount of different holes they have to be completely labelled, using the Hoshen–Kopelman algorithm [18] *for natural boundaries*: each site is not only checked for its upper and left single neighbours, but also for the complete upper and left neighbourhood. After this check in a lot of cases, all sites of the whole area with a certain number have to be renumbered. Therefore, the computing time of that algorithm normally enlarges with the fourth power of the cluster side length and it takes too much time to analyse clusters with more than 1024×1024 pixels. Using the following two-step algorithm this limit has been overcome. As shown in figure 6, the square area of the cluster is firstly divided into smaller squares like a chessboard. By trial and error we found an 8×8 partitioning to be the most effective for clusters larger than 1024×1024. Next, the Hoshen–Kopelman analysis is applied for each partial square *independently*. Dividing the integer number space into non-overlapping ranges guarantees an independent numbering for each square. Note, that in this first step nearly all of the small- and medium-sized holes are finally renumbered, if they are completely located within the area of one partial square. We have got advantage by additionally using a message-passing interface with one CPU for each square, which results in saving considerable time for this step of computation.

The final step of the analysis is to apply the Hoshen–Kopelman algorithm again, *but only* for the pixels at the top and left edge of each partial square, beginning at the upper left and ending at the lower right corner of the chessboard. The most complicated situation to be tackled now is, that two holes of an upper or left subsquare, so far *not* connected within this subsquare, can now be linked by a new hole in the actual subsquare, in the case of natural borderlines already by one single empty pixel as shown in the detail of figure 6. In that special case two of three holes must be renumbered simultaneously. Nevertheless, in this step of computation, the number of pixels analysed is only a *linear* function of the cluster side length and, in most cases, only holes that are in contact with a hole on the other side of a borderline are renumbered.

As you will see in subsection 5.2, it was necessary to handle clusters up to 4096×4096 pixels, and it is the method described here, which enables this handling. The method reduces the computation time for *one* 4096×4096 cluster from a couple of days to about 1 h.

5.2. Results

For 500 clusters with 1024×1024 pixels each, figure 7 shows the average perimeter of holes versus area. The hole fractal dimensions have been calculated from the results of regression, equation (1), as in chapter 4.2.

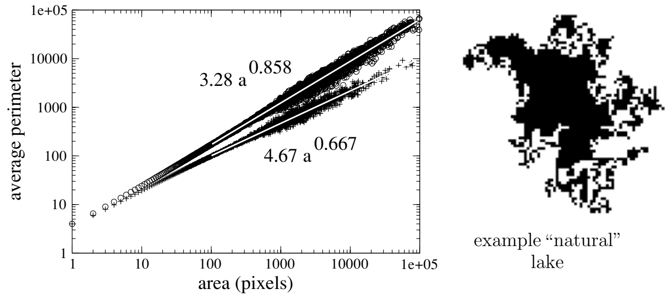

Figure 7. Average perimeter for each area for random Sierpinski carpets 1024 × 1024, together with the results of regressions (white lines), circles: natural borderlines, crosses: nearest-neighbour-borderlines.

Table 1. Results for hole dimensions (nat.) in 50 percolation clusters 4096 × 4096, allowing only holes with a minimal size.

Hole size taken into consideration (pixels)	Number of holes	Calculated fractal dimension
⩾100	342 918	1.723
⩾1000	39 714	1.730
⩾10 000	4 469	1.744

Table 2. Results for hull and hole dimensions calculated for natural borderlines (nat) and nearest neighbour connections (nn). (500 clusters 1024 × 1024, bold numbers: 50 clusters 4096 × 4096).

	Random SC (nat)	Random SC (nn)	Percolation cluster (nat)	Percolation cluster (nn)
Hull dimension	1.48	1.37	1.76	1.36
Hole dimension	1.474	1.364	**1.744**	**1.346**

In an early stage of collecting data, we supposed a weak underestimation of the hole dimensions of percolation clusters due to the finite size and granularity of the structures. To analyse this systematic error more precisely, we performed an additional computation series, exemplarily with much larger clusters containing 4096 × 4096 pixels and using the method described in subsection 5.1. This cluster size allows us to extract data for particularly large holes. Table 1 shows clearly that the lake-shore fractal dimension increases by excluding small holes. Moreover, it seems that the value tends to the theoretical result 7/4. It turned out that this effect appears specifically for percolation clusters with natural borderlines.

The dimension for nearest neighbour connected holes only slightly increases in analysing 4096 × 4096 clusters, due to the too rare appearance of holes with an area greater than 1000 pixels. We have taken the mean value for areas greater or equal 100 as the final result. All numerical results for fractal dimensions in random Sierpinski carpets and percolation clusters are summarized in section 6, table 2. The hull dimension results are in a very good agreement with earlier work [19, 20].

Quite interesting properties have been found for 'lakes with islands'. After its primary generation, the percolation cluster has islands within its internal lakes, again second-generation lakes within these, and so on. For all calculations discussed so far, these islands, not belonging

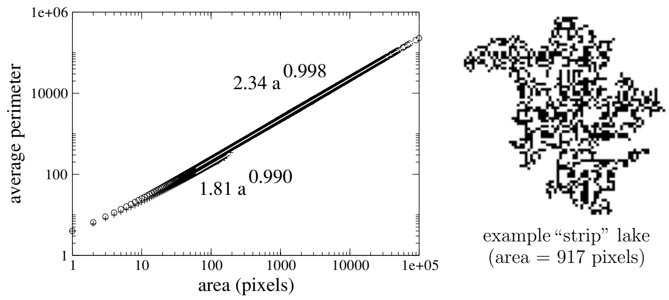

Figure 8. Average perimeter for each area for random percolation clusters *with islands*, 1024×1024, together with the results of regressions (white lines), circles: natural borderlines, crosses: nearest-neighbour-borderlines.

to the cluster itself, had been deleted. Otherwise, in case one does not delete these islands and includes their boundaries when calculating the 'lake shore' dimension, the results are quite different. Including all the boundaries of these islands reduces the areas and increases the perimeters. As shown in figure 8, we can easily obtain for this case a fractal dimension up to 1.996, which allows us to predict it to have generally a value of $(2 - \varepsilon)(0 < \varepsilon \ll 1)$ for large lakes with islands. This is of course very different from the 'hull dimension'. Indeed, an analysis of the pixel properties indicates that for these lakes nearly all pixels (up to about 97%) touch the perimeter. Therefore, it is evident that the islands 'deform' the lakes to narrow strips, either with a strongly restricted length when only the rigorous nearest neighbour connections are allowed (maximal length found: 190 pixels), or with considerable lengths if the natural connection rules are applied. These linear strips are uniformly and randomly distributed inside the cluster, like a two-dimensional network of lines with a uniform mean density. Consequently, their fractal dimension is nearly equal to that of an Euclidean plane. An example for such a kind of lake is shown (magnified) in figure 8, only 23 pixels (out of 917) of this lake are not part of the perimeter.

6. Conclusions

Table 2 shows that hull and hole dimensions for random fractals do agree within limits of computational error. The standard errors determined in averaging out the samples of each kind of clusters are all close to ± 0.02 for the hull data and ± 0.006 for the hole data, respectively.

Therefore, we conclude from table 2 that our conjecture—external and internal perimeters should have the same fractal dimension—appears to hold well for random fractals. For deterministic SC, we have shown that this is an exact result.

Though this conclusion seems very reasonable, it is by no means obvious. One may better realize that it is non-trivial from the following considerations. It is perfectly possible for a compact non-fractal object to have a fractal boundary as for example in Eden clusters [21] or the triadic Koch island [1]. These examples do not have internal holes. Our present study shows that as soon as internal holes appear naturally through the fractal generation rule, they should have the same dimension as the outer boundary. Again, one can also think of mass fractals with a smooth hull, SC of this class are easy to construct, e.g. those in Franz *et al* [5] with $d_p = 1$. These patterns have internal holes of all sizes (otherwise they could not be mass fractals), but smoothness of the external boundary implies that the internal holes cannot have fractal interfaces, according to the findings presented here.

A random walker walking along the hull or a lake shore returns to its starting point and may not be able to distinguish whether it has traversed an internal or the external boundary. But this does not mean that hull and lake shores are equivalent and indistinguishable. If the walker compares the sets of lattice sites bounded by the perimeter for all such walks, all the sets, except one will be disjoint with no common point. The remaining one will include all the others and this corresponds to the hull, so in principle the hull is distinguishable from the lake perimeters.

The fractal dimension of the internal boundary of a cavity in a porous fractal [5] has not received much attention, though there are many studies related to the hull of a percolation cluster [19, 20]. Now we have demonstrated that for deterministic as well as random fractals, where the internal cavities and external boundary are formed by the same iteration rule or the same natural growth process, the internal and external perimeters scale similarly. We can expect to see this in practical cases, such as vacuum deposition on a substrate or electrodeposition from an electrolyte solution.

The result may not hold for example, in the case of a piece of porous rock broken off from a large sample. In this case, the outer surface is formed by the process of fracture, whereas the internal holes are a result of the sedimentation and diagenesis processes which created the rock [22, 23]. So these new results are of practical importance and convey some information about how the surfaces are formed. Furthermore, for mathematical fractals the result may have a deeper topological implication.

Acknowledgments

S Tarafdar is grateful to the DFG, especially Sonderforschungsbereich 393 'Numerische Simulation auf massiv parallelen Rechnern', for funding a visit to Chemnitz.

References

[1] Mandelbrot B B 1977 *Fractals: Form, Chance and Dimensions* (San Francisco: Freeman)
[2] Vicsek T 1992 *Fractal Growth Phenomena* (Singapore: World Scientific)
[3] Bunde A and Havlin S 1994 *Fractals in Science* (Berlin: Springer)
[4] Ziff R M, Cummings P T and Stells G 1986 *J. Phys. A: Math. Gen.* **17** 3009
[5] Franz A, Schulzky C, Tarafdar S and Hoffmann K H 2001 *J. Phys. A: Math. Gen.* **34** 8751
[6] Stauffer D and Aharony A 1994 *Introduction to Percolation Theory* (London: Taylor and Francis)
[7] Jarai A A 2003 *Ann. Probab.* **31** 444
[8] Krapinsky P L and Derrida B 2004 *Physica* A **340** 714
[9] Monceau P and Hsian P-Y 2004 *Phys. Lett.* A **332** 310
[10] Martin J E and Keefer K D 1985 *J. Phys. A: Math. Gen.* **18** L625
[11] Nagatani T 1986 *J. Phys. A: Math. Gen.* **19** L1079
[12] Perreau M, Peiro J and Berthier S 1996 *Phys. Rev.* E **54** 4590
[13] Reiss F D A A 1996 *J. Phys. A: Math. Gen.* **29** 7803
[14] Kim M H, Yoon D H and Kim I M 1993 *J. Phys. A: Math. Gen.* **26** 5655
[15] Lin B and Yang Z R 1986 *J. Phys. A: Math. Gen.* **19** L49
[16] Bonnier B, Leroyer Y and Meyers C 1988 *Phys. Rev.* B **37** 5205
[17] Ben Avraham D and Havlin S 2000 *Diffusion and Reactions in Fractal and Disordered Systems* (Cambridge: Cambridge University Press)
[18] Hoshen J and Kopelman R 1976 *Phys. Rev.* B **14** 3428
[19] Grossman T and Aharony A 1986 *J. Phys. A: Math. Gen.* **19** L745
[20] Saleur H and Duplantier B 1987 *Phys. Rev. Lett.* **58** 2325
[21] Freche P, Stauffer D and Stanley H E 1985 *J Phys. A: Math. Gen.* **18** L1163
[22] Pettijohn F J 1984 *Sedimentary Rocks* (New Delhi: CBS Publishers)
[23] Dutta T and Tarafdar S 2003 *J. Geophys. Res.* **108** 2062