Chapter 9
Generalized Dimensions

A multifractal is a fractal that cannot be characterized by a single fractal dimension such as the box counting dimension. The infinite number of fractal dimensions needed in general to characterize a multifractal are known as generalized dimensions. Generalized dimensions of geometric multifractals were proposed independently in 1983 by Grassberger [20] and by Hentschel and Procaccia [25]. They have been intensely studied (e.g., [21, 40, 61]) and widely applied (e.g., [39, 59]). Given N points from a geometric multifractal, e.g., the strange attractor of a dynamical system [9, 41], the generalized dimension D_q defined in [20, 25] is computed from a set of box sizes. For box size s, we cover the N points with a grid of boxes of linear size s, compute the fraction $p_j(s)$ of the N points in box B_j of the grid, discard any box for which $p_j(s) = 0$, and compute the partition function value

$$Z_q\big(\mathscr{B}(s)\big) \equiv \sum_{B_j \in \mathscr{B}(s)} [p_j(s)]^q \,, \tag{9.1}$$

where $\mathscr{B}(s)$ is the set of non-empty grid boxes, of linear size s, used to cover the N points. For $q \geq 0$ and $q \neq 1$, the generalized dimension D_q defined in [20, 25] of the geometric multifractal is

$$D_q \equiv \frac{1}{q-1} \lim_{s \to 0} \frac{\log Z_q\big(\mathscr{B}(s)\big)}{\log s} \,. \tag{9.2}$$

When $q = 0$, this computation yields the box counting dimension d_B, so $D_0 = d_B$. When $q = 1$, after applying L'Hôpital's rule we obtain the information dimension d_I [13], so $D_1 = d_I$. When $q = 2$, we obtain the correlation dimension d_C [23], so $D_2 = d_C$.

Generalized dimensions of a complex network were studied in [15, 34, 48, 49, 58, 67, 68]. Several of these studies employ the *sandbox method*, which we discuss

E. Rosenberg, *A Survey of Fractal Dimensions of Networks*, SpringerBriefs in Computer Science, https://doi.org/10.1007/978-3-319-90047-6_9

at the end of this chapter. The method of [67] for computing D_q for $\mathbb{G}$ is the following. For a range of s, compute a minimal s-covering $\mathscr{B}(s)$. For $B_j \in \mathscr{B}(s)$, define $p_j(s) \equiv N_j(s)/N$, where $N_j(s)$ is the number of nodes in B_j. For $q \in \mathbb{R}$, use (9.1) to compute $Z_q\big(\mathscr{B}(s)\big)$. (In [67], which uses a randomized box counting heuristic, $Z_q\big(\mathscr{B}(s)\big)$ is the average partition function value, averaged over 200 random orderings of the nodes.) Typically, D_q is computed only for a small set of q values, e.g., integer q in $[0, 10]$ or integer q in $[-10, 10]$. Then $\mathbb{G}$ has the generalized dimension D_q (for $q \neq 1$) if for some constant c and for some range of s we have

$$\log Z_q\big(\mathscr{B}(s)\big) \approx (q-1)D_q \log(s/\Delta) + c\,. \tag{9.3}$$

However, as shown in [48], this definition is ambiguous, since different minimal s-coverings can yield different values of D_q.

Example 9.1 Consider again the *chair* network of Fig. 8.2, which shows two minimal 3-coverings and a minimal 2-covering. Choosing $q = 2$, for the covering $\widetilde{\mathscr{B}}(3)$ from (9.1) we have $Z_2\big(\widetilde{\mathscr{B}}(3)\big) = (\frac{3}{5})^2 + (\frac{2}{5})^2 = \frac{13}{25}$, while for $\widehat{\mathscr{B}}(3)$ we have $Z_2\big(\widehat{\mathscr{B}}(3)\big) = (\frac{4}{5})^2 + (\frac{1}{5})^2 = \frac{17}{25}$. For $\mathscr{B}(2)$ we have $Z_2\big(\mathscr{B}(2)\big) = 2(\frac{2}{5})^2 + (\frac{1}{5})^2 = \frac{9}{25}$. If we use $\widetilde{\mathscr{B}}(3)$ then from (9.3) and the range $s \in [2, 3]$ we obtain

$$D_2 = \left(\log\frac{13}{25} - \log\frac{9}{25}\right)/(\log 3 - \log 2) \approx 0.907\,.$$

If instead we use $\widehat{\mathscr{B}}(3)$ and the same range of s we obtain

$$D_2 = \left(\log\frac{17}{25} - \log\frac{9}{25}\right)/(\log 3 - \log 2) \approx 1.569\,.$$

Thus the method of [67] can yield different values of D_2 depending on the minimal covering selected. □

To devise a computationally efficient method for selecting a unique minimal covering, first consider the maximal entropy criterion described in Chap. 8. It is well known that entropy is *maximized* when all the probabilities are equal. A partition function is *minimized* when the probabilities are equal. To formalize this idea, for integer $J \geq 2$, let **P(q)** denote the continuous optimization problem: minimize $\sum_{j=1}^{J} p_j^q$ subject to $\sum_{j=1}^{J} p_j = 1$ and $p_j \geq 0$ for each j. It is proved in [48] that for $q > 1$, the solution of **P(q)** is $p_j = 1/J$ for each j. Applying this result to $\mathbb{G}$, minimizing $Z_q\big(\mathscr{B}(s)\big)$ over all minimal s-coverings of $\mathbb{G}$ yields a minimal s-covering for which all the probabilities $p_j(s)$ are, to the extent possible, equalized. Since $p_j(s) = N_j(s)/N$, equal box probabilities means that all boxes in the minimal s-covering have the same number of nodes. The following definition [48] of an (s, q) minimal covering, for use in computing D_q, is analogous to the definition in [47] of a maximal entropy minimal s-covering, for use in computing d_I.

Definition 9.1 For $q \in \mathbb{R}$, the covering $\mathscr{B}(s)$ of $\mathbb{G}$ is an (s, q) minimal covering if (i) $\mathscr{B}(s)$ is a minimal s-covering and (ii) for any other minimal s-covering $\widetilde{\mathscr{B}}(s)$ we have $Z_q\big(\mathscr{B}(s)\big) \leq Z_q\big(\widetilde{\mathscr{B}}(s)\big)$. □

It is easy to modify any box counting method (in a manner analogous to Procedure 8.1) to compute an (s, q) minimal covering for a given s and q. However, this approach to eliminating ambiguity in the computation of a minimal s-covering is not particularly attractive, since it requires computing an (s, q) minimal covering for each value of q for which we wish to compute D_q. A better approach to resolving this ambiguity is to compute a lexico minimal summary vector [48], which summarizes an s-covering $\mathscr{B}(s)$ by the point $x \in \mathbb{R}^J$, where $J \equiv B(s)$, where $x_j = N_j(s)$ for $1 \leq j \leq J$, and where $x_1 \geq x_2 \geq \cdots \geq x_J$. (We use *lexico* instead of the longer *lexicographically*.) The vector x does not specify all the information in $\mathscr{B}(s)$; in particular, $\mathscr{B}(s)$ specifies exactly which nodes belong to each box, while x specifies only the number of nodes in each box. The notation $x = \sum \mathscr{B}(s)$ signifies that x summarizes the s-covering $\mathscr{B}(s)$ and that $x_1 \geq x_2 \geq \cdots \geq x_J$. For example, if $N = 37$, $s = 3$, and $B(3) = 5$, we might have $x = \sum \mathscr{B}(3)$ for $x = (18, 7, 5, 5, 2)$. However, we cannot have $x = \sum \mathscr{B}(3)$ for $x = (7, 18, 5, 5, 2)$ since the components of x are not ordered correctly. If $x = \sum \mathscr{B}(s)$ then each x_j is positive, since x_j is the number of nodes in box B_j. The vector $x = \sum \mathscr{B}(s)$ a called a *summary* of $\mathscr{B}(s)$. By "x is a summary" we mean x is a summary of $\mathscr{B}(s)$ for some $\mathscr{B}(s)$. For $x(s) = \sum \mathscr{B}(s)$ and $q \in \mathbb{R}$, define

$$Z\big(x(s), q\big) \equiv \sum_{B_j \in \mathscr{B}(s)} \left(\frac{x_j(s)}{N}\right)^q . \tag{9.4}$$

Thus for $x(s) = \sum \mathscr{B}(s)$ we have $Z\big(x(s), q\big) = Z_q\big(\mathscr{B}(s)\big)$, where $Z_q\big(\mathscr{B}(s)\big)$ is defined by (9.1).

Let $x \in \mathbb{R}^K$ for some positive integer K. Let $right(x) \in \mathbb{R}^{K-1}$ be the point obtained by deleting the first component of x. For example, if $x = (18, 7, 5, 5, 2)$ then $right(x) = (7, 5, 5, 2)$. Similarly, we define $right^2(x) \equiv right\big(right(x)\big)$, so $right^2(7, 7, 5, 2) = (5, 2)$. Let $u \in \mathbb{R}$ and $v \in \mathbb{R}$ be numbers. We say that $u \succeq v$ (in words, u is *lexico* greater than or equal to v) if ordinary inequality holds, that is, $u \succeq v$ if $u \geq v$. Thus $6 \succeq 3$ and $3 \succeq 3$. Now let $x \in \mathbb{R}^K$ and $y \in \mathbb{R}^K$. We define lexico inequality recursively: we say that $y \succeq x$ if either (i) $y_1 > x_1$ or (ii) $y_1 = x_1$ and $right(y) \succeq right(x)$. For example, for $x = (9, 6, 5, 5, 2)$, $y = (9, 6, 4, 6, 2)$, and $z = (8, 7, 5, 5, 2)$, we have $x \succeq y$ and $x \succeq z$ and $y \succeq z$.

Definition 9.2 Let $x = \sum \mathscr{B}(s)$. Then x is *lexico minimal* if (i) $\mathscr{B}(s)$ is a minimal s-covering and (ii) if $\widetilde{\mathscr{B}}(s)$ is a minimal s-covering distinct from $\mathscr{B}(s)$ and $y = \sum \widetilde{\mathscr{B}}(s)$ then $y \succeq x$. □

The following two theorems are proved in [48].

Theorem 9.1 *For each s there is a unique lexico minimal summary.*

Theorem 9.2 *Let* $x = \sum \mathscr{B}(s)$*. If* x *is lexico minimal then* $\mathscr{B}(s)$ *is* (s, q) *minimal for all sufficiently large* q*.*

Analogous to Procedure 8.1, Procedure 9.1 below shows how, for a given s, the lexico minimal $x(s)$ can be computed by a simple modification of whatever box counting method is used to compute a minimal s-covering.

Procedure 9.1 Let $\mathscr{B}_{\min}(s)$ be the best s-covering obtained over all executions of whatever box counting method is utilized. Suppose we have executed box counting some number of times, and stored $\mathscr{B}_{\min}(s)$ and $x_{\min}(s) = \sum \mathscr{B}_{\min}(s)$, so $x_{\min}(s)$ is the current best estimate of a lexico minimal summary vector. Now suppose we execute box counting again, and generate a new s-covering $\mathscr{B}(s)$ using $B(s)$ boxes. Let $x = \sum \mathscr{B}(s)$. If $B(s) < B_{\min}(s)$, or if $B(s) = B_{\min}(s)$ and $x_{\min}(s) \succeq x$, then set $\mathscr{B}_{\min}(s) = \mathscr{B}(s)$ and $x_{\min}(s) = x$. □

Procedure 9.1 shows that the only additional steps, beyond the box counting method itself, needed to compute $x(s)$ are lexicographic comparisons, and no evaluations of the partition function $Z_q\big(\mathscr{B}(s)\big)$ are required. By Theorems 9.1 and 9.2, the summary vector $x(s)$ is unique and also "optimal" (i.e., (s, q) minimal) for all sufficiently large q. Thus an attractive way to resolve ambiguity in the choice of minimal s-coverings is to compute $x(s)$ for a range of s and use the $x(s)$ vectors to compute D_q, using Definition 9.3 below.

Definition 9.3 For $q \neq 1$, the complex network $\mathbb{G}$ has the generalized dimension D_q if for some constant c and for some range of s we have

$$\log Z\big(x(s), q\big) \approx (q-1)D_q \log(s/\Delta) + c\,, \tag{9.5}$$

where $x(s) = \sum \mathscr{B}(s)$ is lexico minimal. □

Example 9.2 (Continued) Consider again the *chair* network of Fig. 8.2. Choose $q = 2$. For $s = 2$ we have $x(2) = \sum \mathscr{B}(2) = (2, 2, 1)$ and $Z\big(x(2), 2\big) = \frac{9}{25}$. For $s = 3$ we have $\widetilde{x}(3) = \sum \widetilde{\mathscr{B}}(3) = (3, 2)$ and $Z\big(\widetilde{x}(3), 2\big) = \frac{13}{25}$. Over the range $s \in [2, 3]$, from Definition 9.3 we have $D_2 = \log(13/9)/\log(3/2) \approx 0.907$. For this network, not only is the value of D_q dependent on the minimal s-covering selected, but even the overall shape of the D_q vs. q curve depends on the minimal s-covering selection. For $x(2) = (2, 2, 1)$ we have

$$Z\big(x(2), q\big) = 2\left(\frac{2}{5}\right)^q + \left(\frac{1}{5}\right)^q .$$

For $\widetilde{x}(3) = (3, 2)$ we have

$$Z\big(\widetilde{x}(3), q\big) = \left(\frac{3}{5}\right)^q + \left(\frac{2}{5}\right)^q .$$

Over the range $s \in [2, 3]$, from (9.5) we have

$$\widetilde{D}_q \equiv \left(\frac{1}{q-1}\right)\left(\frac{\log\left(\frac{3^q+2^q}{5^q}\right) - \log\left(\frac{(2)(2^q)+1}{5^q}\right)}{\log(3/\Delta) - \log(2/\Delta)}\right) = \frac{\log\left(\frac{3^q+2^q}{(2)(2^q)+1}\right)}{\log(3/2)(q-1)} . \quad (9.6)$$

If for $s = 3$ we instead choose the covering $\widehat{\mathscr{B}}(3)$ then for $\widehat{x}(3) = (4, 1)$ we have

$$Z\big(\widehat{x}(3), q\big) = \left(\frac{4}{5}\right)^q + \left(\frac{1}{5}\right)^q .$$

Again over the range $s \in [2, 3]$, but now using $\widehat{x}(3)$ instead of $\widetilde{x}(3)$, we obtain

$$\widehat{D}_q \equiv \left(\frac{1}{q-1}\right)\left(\frac{\log\left(\frac{4^q+1^q}{5^q}\right) - \log\left(\frac{(2)(2^q)+1}{5^q}\right)}{\log(3/\Delta) - \log(2/\Delta)}\right) = \frac{\log\left(\frac{4^q+1}{(2)(2^q)+1}\right)}{\log(3/2)(q-1)} . \quad (9.7)$$

Figure 9.1 plots $\widetilde{D}_q$ vs. q, and $\widehat{D}_q$ vs. q over the range $0 \le q \le 15$. Neither curve is monotone non-increasing: the $\widetilde{D}_q$ curve (corresponding to the lexico minimal summary vector $\widetilde{x}(3) = (3, 2)$) is unimodal, with a local minimum at $q \approx 4.1$, and the $\widehat{D}_q$ curve is monotone increasing. □

The fact that neither curve in Fig. 9.1 is monotone non-increasing is remarkable, since it is well known that for a geometric multifractal, the D_q vs. q curve is monotone non-increasing [20]. The shape of the D_q vs. q curve will be explored further in Chap. 10. We next show that the $x(s)$ summary vectors can be used to compute $D_\infty \equiv \lim_{q\to\infty} D_q$. Let $x(s) = \sum \mathscr{B}(s)$ be lexico minimal, and let $x_1(s)$ be the first element of $x(s)$. It is proved in [48] that

$$\log\left(\frac{x_1(s)}{N}\right) \approx D_\infty \log\left(\frac{s}{\Delta}\right) . \quad (9.8)$$

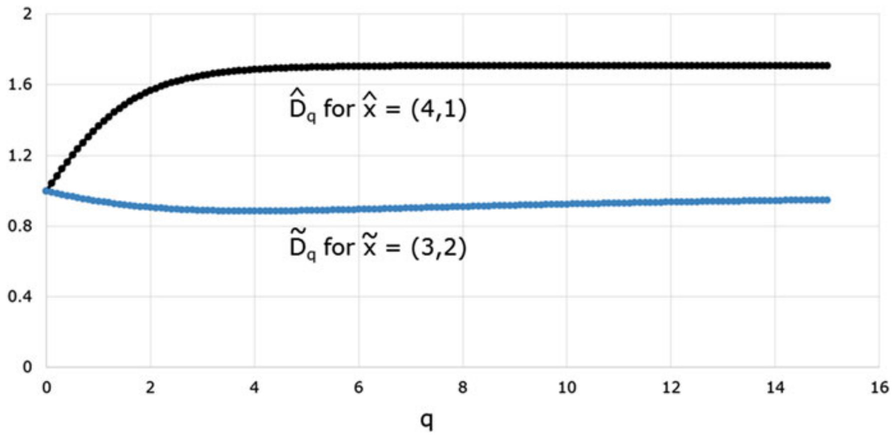

Fig. 9.1 Two plots of the generalized dimensions for the *chair* network

We can use (9.8) to compute D_∞ without having to compute any partition function values. It is well known [41] that, for geometric multifractals, D_∞ corresponds to the densest part of the fractal. Similarly, (9.8) shows that, for a complex network, D_∞ is the factor that relates the box size s to $x_1(s)$, the number of nodes in the box in the lexico minimal s-covering for which $p_j(s)$ is maximal.

To conclude this chapter, we consider the *sandbox method* for approximating D_q. The sandbox method, originally designed to compute D_q for geometric multifractals obtained by simulating diffusion-limited aggregation on a lattice [64, 65, 71], overcomes a well-recognized [1] limitation of using box counting to compute generalized dimensions: spurious results can be obtained for $q \ll 0$. This will happen if some box probability p_j is close to zero, for then when $q \ll 0$ the term p_j^q will dominate the partition sum $\sum_j p_j^q$. The sandbox method has also been shown to be more accurate than box counting for geometric fractals with known theoretical dimensions [62]. To describe the sandbox method, note that for a geometric multifractal for which D_q exists, by (9.1) and (9.2) we have, as $s \to 0$,

$$Z_q\big(\mathscr{B}(s)\big) = \sum_{B_j \in \mathscr{B}(s)} p_j^q(s) = \sum_{B_j \in \mathscr{B}(s)} p_j(s)[p_j(s)]^{q-1} \sim s^{(q-1)D_q} .$$

The sandbox method approximates $\sum_{B_j \in \mathscr{B}(s)} p_j^q(s)$ as follows [62]. Let $\widetilde{\mathbb{N}}$ be a randomly chosen subset of the N points and define $\widetilde{N} \equiv |\widetilde{\mathbb{N}}|$. With $M(n,r)$ defined by (7.1) and (7.2), define

$$avg\big(p^{q-1}(r)\big) \equiv \frac{1}{\widetilde{N}} \sum_{n \in \widetilde{\mathbb{N}}} \left(\frac{M(n,r)}{N}\right)^{q-1} , \tag{9.9}$$

where the notation $avg\big(p^{q-1}(r)\big)$ is chosen to make it clear that this average uses equal weights of $1/\widetilde{N}$. Let L be the linear size of the lattice. The essence of the sandbox method is the approximation, for $r \ll L$,

$$avg\big(p^{q-1}(r)\big) \sim (r/L)^{(q-1)D_q} . \tag{9.10}$$

Note that $\sum_{B_j \in \mathscr{B}(s)} p_j^q(s)$ is a sum over the set of non-empty grid boxes, and the weight applied to $[p_j(s)]^{q-1}$ is $p_j(s)$. In contrast, $avg\big(p^{q-1}(r)\big)$ is a sum over a randomly selected set of sandpiles, and the weight applied to $\big(M(n,r)/N\big)^{q-1}$ is $1/\widetilde{N}$. Since the $\widetilde{N}$ sandpile centers are chosen from the N points using a uniform distribution, the sandpiles may overlap. Because the sandpiles may overlap, and the sandpiles do not necessarily cover all the N points, in general $\sum_{n \in \widetilde{\mathbb{N}}} M(n,r) \neq N$, and we cannot regard the values $\{M(n,r)/N\}_{n \in \widetilde{\mathbb{N}}}$ as a probability distribution. Let β be the spacing between adjacent lattice positions (e.g., between adjacent horizontal and vertical positions for a lattice in $\mathbb{R}^2$).

Definition 9.4 For $q \neq 1$, the *sandbox dimension function* [62] of order q is the function of r defined for $\beta \le r \ll L$ by

$$D_q^{sandbox}(r/L) \equiv \frac{1}{q-1} \frac{\log avg\big(p^{q-1}(r)\big)}{\log(r/L)} \,. \tag{9.11}$$

□

For a given $q \neq 1$ and lattice size L, the sandbox dimension function does not define a single sandbox dimension, but rather a range of sandbox dimensions, depending on r. It is not meaningful to define $\lim_{r \to 0} D_q^{sandbox}(r/L)$, since r cannot be smaller than the spacing β between lattice points. In practice, for a given q and L, a single value $D_q^{sandbox}$ of the sandbox dimension of order q is typically obtained by computing $D_q^{sandbox}(r/L)$ for a range of r values, and finding the slope of the $\log avg\big(p^{q-1}(r)\big)$ vs. $\log(r/L)$ curve. The estimate of $D_q^{sandbox}$ is $1/(q-1)$ times this slope.

The sandbox method was applied to complex networks in [34]. The box centers are randomly selected nodes. There is no firm rule in [34] on the number of random centers to pick: they use $\widetilde{N} \equiv |\widetilde{\mathbb{N}}| = 1000$ random nodes, but suggest that $\widetilde{N}$ can depend on N. For a given $q \neq 1$, they compute $avg\big(p^{q-1}(r)\big)$ for a range of r values. Adapting (9.10) to a complex network $\mathbb{G}$, for $r \ll \Delta$ we have

$$\log avg\big(p^{q-1}(r)\big) \sim (q-1) D_q^{sandbox} \log(r/\Delta) \,. \tag{9.12}$$

In [34], linear regression is applied to (9.12) to compute $D_q^{sandbox}$.

The sandbox method was applied to undirected weighted networks in [58]. The calculation of the sandbox radii in [58] is similar to the selection of box sizes discussed in Sect. 3.2.