Tommaso Bolognesi
CNUCE
36 Via Santa Maria
56100 Pisa, Italy

Automatic Composition: Experiments with Self-Similar Music

Introduction

This paper describes some automatic composition experiments based on stochastic processes. Usually the basic questions raised as a premise to experiments of this kind (or, often, as a conclusion) sound like these:

- Can we use random-number generators in a musically meaningful way, producing something better than *white melodies*, in which pitches are mutually independent and uniformly distributed?
- Does the large variety of different variables and stochastic processes studied mathematically correspond to a large variety of perceptual experiences?
- Can we "tune" the parameters of a stochastic process to generate musical macrostructures in the same way we tune, at the microstructural level (timbre), the parameters of the various sound-synthesis models?
- How far can we go using exclusively stochastic processes, without embedding in our compositional algorithms any kind of formalized musical grammar?

Several random variables, their distribution functions, and some of their possible musical applications are discussed by Lorrain (1980). (See Lorrain's paper for discussion of the concepts of *random variable*, *density*, and *distribution function*.) An interesting way of "tuning" a statistical distribution of points in time (e.g., to generate rhythmic patterns) that allows a continuous transformation from a Poisson distribution into a sequence of equally spaced points has been suggested by Myhill (1979). Myhill's approach requires that only a single parameter be manipulated.

In the composition experiments discussed here, we use exclusively stochastic processes. We only allow for global control over the process (tuning) by means, essentially, of a single parameter. In particular, these processes, as opposed to "white" random sources, tend to organize themselves into hierarchical patterns. This is certainly a perceivable and musically meaningful property.

Programs written to simulate such processes and produce music have been implemented in the framework of the real-time TAU2-TAUMUS system developed at Pisa (Bertini et al. 1980).

Starting Point

The starting point of this work was a surprising experimental result obtained by Voss and Clarke (1978). Their analysis focused on the fluctuations in time of two acoustic parameters: the *audio power* $V^2(t)$ ($V[t]$ being the audio signal), which varies monotonically with the *loudness* of the music, and the *rate of zero crossings* $Z(t)$. $Z(t)$ is the number of times the audio signal $V(t)$ crosses the time axis in a time unit. $Z(t)$ roughly follows the *melody*. A statistical characterization of the behavior of these parameters is given, in the frequency (f) domain, by their power spectra $S_v^2(f)$ and $S_z(f)$. These represent the mean-squared variation of $V^2(t)$ and $Z(t)$ respectively, in a unit bandwidth centered on the frequency f. Information about timbral (microstructural) properties of a given piece is commonly obtained by looking at the power spectrum of a short portion of the audio signal $V(t)$ (which is a rapidly fluctuating quantity), in the range, say, of 16 Hz to 20 KHz. Conversely, Clarke and Voss's idea was to plot $S_v^2(f)$ and $S_z(f)$ down to very low frequencies (about 10^{-3} Hz), in order to characterize fluctuations of loud-

This is a revised version of a lecture presented at the Third Colloquium on Musical Informatics held at Padua University, Padua, Italy.

Computer Music Journal, Vol. 7, No. 1,
Spring 1983, 0148-9267/83/010025-12 $04.00/0,

Code Listing 1. Voss's algorithm for generating an approximation of 1/f digitized noise.

```
for i = 0 to n − 1 do oldbit [i] = 1
for K = 0 to 2^n − 1 do
 (bit [0], bit [1] . . . bit [n − 1]) ← binary representation of K
 sum = 0
 for i = 0 to n − 1 do
  if bit [i] ≠ oldbit [i] do
  die [i] ← random (1, MAX)
  sum ← sum + die [i]
  oldbit [i] ← bit [i]
 output sum
```

ness and melody on a very large time scale and detect macrostructural properties of music. The result they obtained is that a large number of musical styles, from Bach to rock music, show to some extent a "$1/f$ behavior"; that is, spectra proportional to $1/f$ in some interval at very low frequencies. This observation suggested the use of stochastic processes with a $1/f$ spectrum (pink noise) rather than a constant spectrum (white noise) to control time fluctuations of parameters like frequency and loudness in automatic composition experiments.

The properties of the music generated by this algorithm show the relation between $1/f$ spectra and hierarchical (yet stochastic) structures. More precisely, the stochastic melodies generated by this algorithm and by some variants of it (described later) show what may probably be considered the most primitive form of hierarchy: *self-similarity*. A self-similar structure is one whose parts recursively repeat the whole structure, and this immediately implies a hierarchy with an infinite number of levels. (Many self-similar geometric structures may be found in Mandelbrot's work [1977].) The idea of self-similarity is so fascinating on its own that we may set aside $1/f$ spectra and make it a new starting point for our experiments. Later, in the section called Self-Similar Music, we take a walk, or better, a *flight* (as will become clear) into the domain of self-similar stochastic music.

1/f Music

An algorithm used to generate samples of approximately $1/f$ digitized noise, suggested by Voss (Gardner 1978), is given in Code Listing 1. The $2^n - 1$ integers (the variable sum) are generated, each obtained by adding the values of n dice (array die, of n elements). A die is cast by the function call random (1,MAX), which returns a pseudorandom integer x, uniformly distributed in the range $1 \leq x \leq$ MAX. At each iteration of the loop with index K, the binary representation of K (array bit of n elements) is compared with the binary representation of K − 1 (array oldbit): only the m least significant bits may have commuted ($m = 1, 2 \ldots n$), and consequently only the first m dice are cast again, before the values of all dice are added to obtain a new value of sum.

A possible run of the algorithm with three dice ($n = 3$) is illustrated in Fig. 1. Each column corresponds to an iteration over K. Rows are associated with dice. The binary representation of K is given in the upper left corners of the squares in the corresponding column, while bold vertical lines show the pattern of bit commutations, and parentheses indicate permanence of the previous outcome (at the left). For example, to obtain the third output value $14 = 3 + 5 + 6$, we recast die 0 (bit commutation $1 \rightarrow 0$) and die 1 (bit commutation $0 \rightarrow 1$), but we keep the old outcome of die 2, as no bit commutation took place on its corresponding row. If we generate the output values at a constant rate and we think of them as indexes in a given range of pitches, we get a melody (the bold contour of Fig. 2).

Melodic Properties

Each die i is periodically cast every $T_i = 2^i$ time units. The hierarchical way the dice are cast is reflected in some musically interesting properties of

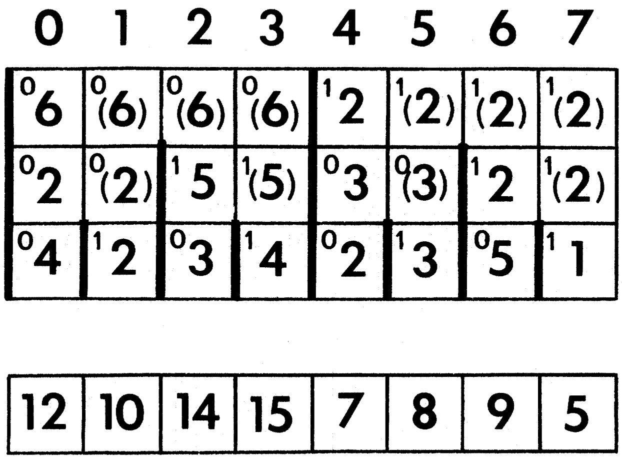


Fig. 1. Casting dice according to Voss's algorithm.

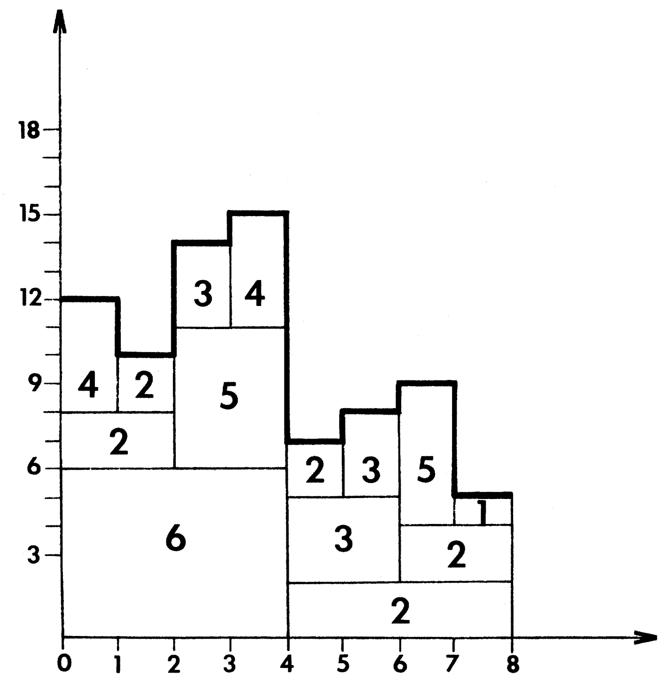


Fig. 2. The melody obtained from the data of Fig. 1.

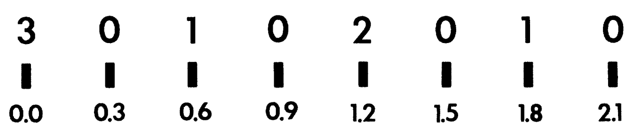


Fig. 3. A natural accentuation in a sequence of pulses.

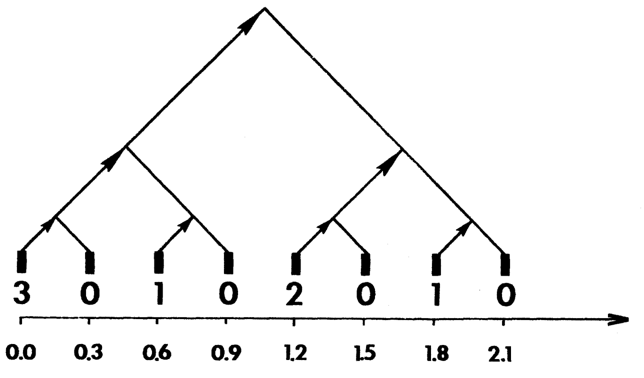
the melodic lines. The pitch interval between two adjacent notes is a random variable depending on the number of dice recast to obtain the second one. This number, in turn, depends on time, as pointed out by bold vertical segments in Fig. 1. The result is a particular statistical behavior that places these melodies in an intermediate region of the hypothetical continuum between two equally monotonous opposite extremes: *brownian melodies*, in which pitch intervals are always small, and *white melodies*, in which pitches are independent of one another (in both cases pitches and pitch intervals are independent of time). By proportioning and alternating in time the behavior of the two extremes, $1/f$ melodies break monotonicity and appear varied to some extent.

From a different point of view, we may observe that melodies are built as a pair of juxtaposed melodic lines, recursively divided into subphrases. Hierarchical levels are as numerous as the dice. Moreover, every phrase at each hierarchical level is statistically similar to its subphrases. As a consequence, the whole melody is statistically similar to all its parts: this is the idea of self-similarity. (The concept of self-similarity is introduced and given a rigorous definition in Mandelbrot's paper [1977], in which a distinction between self-similarity and self-affinity is also made. Nevertheless, in order not to complicate our terminology, the former term will be used even when the latter would be more appropriate.) Theoretically, self-similarity is possible only with an infinite number of hierarchical levels (i.e., of dice), while in the finite case it can only be approximated. Still, this concept seems to fit very well the properties of melodic lines such as the one in Fig. 2.

The musical importance of the hierarchies generated by self-similarity is clear. Let us consider, as an example, the typical disposition of accents, measured on a scale of relative intensity from 0 (low) to 3 (high), in the elementary ryhthmic sequence of Fig. 3.

This accentuation is rooted so deeply in our perceptual habits that it is "perceived" even in sequences of identical pulses and may be easily shown to reflect a hierarchical structure, that of a binary tree (see Fig. 4). Each pulse is associated with a leaf of the tree, and its intensity is obtained by counting the first consecutive right-oriented edges on a path from the leaf toward the root of the tree, as was observed by Lidov and Gabura (1973).

Fig. 4. Hierarchical disposition of accents in a sequence of pulses.



A relation between $1/f$ spectra and hierarchical structures is not only suggested a posteriori by the analysis of the melodies generated by Voss's algorithm, but may also be established on a purely theoretical basis. In order to do this we need to introduce the idea of the *autocorrelation function* of a stochastic process. Fortunately, we do not have to be too rigorous because for many processes the concept has a quite strong intuitive counterpart, and in this context we may rely on that.

For our purposes, a stochastic process $s(t)$ may be defined as a set of *time functions* associated with certain probabilities, in the same way as a random variable is a set of *values* associated with certain probabilities. (We do not "see" many functions in the notation $s(t)$, exactly as we do not "see" many values in the notation x for a random variable.) A basic difference between a random variable and a stochastic process is the time dimension appearing in the latter. If we sample a stochastic process at a given time t_0, we get a random variable $s(t_0)$, that is, the set of values assumed at time t_0 by the functions of the stochastic process (along with their probabilities). The autocorrelation function $R_s(t)$ of a (stationary) stochastic process $s(t)$ represents the correlation between any two samples of the process at a time distance t. A high correlation between $s(t_0)$ and $s(t_0 + t)$, that is, a high value of $R_s(t)$, means the following. When the value taken by the random variable $s(t_0)$ is known, we may predict with high accuracy the value taken by $s(t_0 + t)$, and vice versa. In other words, the function $R_s(t)$ represents the memory of the past after a time t.

Let us now assume that a given stochastic process $s(t)$ has, as an autocorrelation function, the exponential decay

$$R_s(t) = e^{-t/\tau} \qquad (t \geq 0).$$

It is easy to verify that, for any t, $R_s(t + \tau) = (1/e)R_s(t)$. We may then say that τ is a characteristic constant representing the time interval over which $R_s(t)$ gets reduced by a factor $1/e$. More generally, τ represents a characteristic time scale associated with the process. The power spectrum $P_s(f)$ of this process, which is a function in the frequency domain and can be immediately derived mathematically from $R_s(t)$, shows no $1/f$ behavior in any frequency interval. However, Van der Ziel (1950) built a theoretical model of $1/f$ noise by transforming the constant τ into a random variable with density function $g_\tau(x)$ proportional to $1/x$. The expression $g_\tau(x)dx$ represents the probability that τ falls in the interval $(x, x + dx)$. In simpler terms, Van der Ziel obtained a $1/f$ spectrum in a certain frequency interval by assuming, as a fundamental feature of his stochastic process, a whole set of time scales instead of a single constant scale τ.

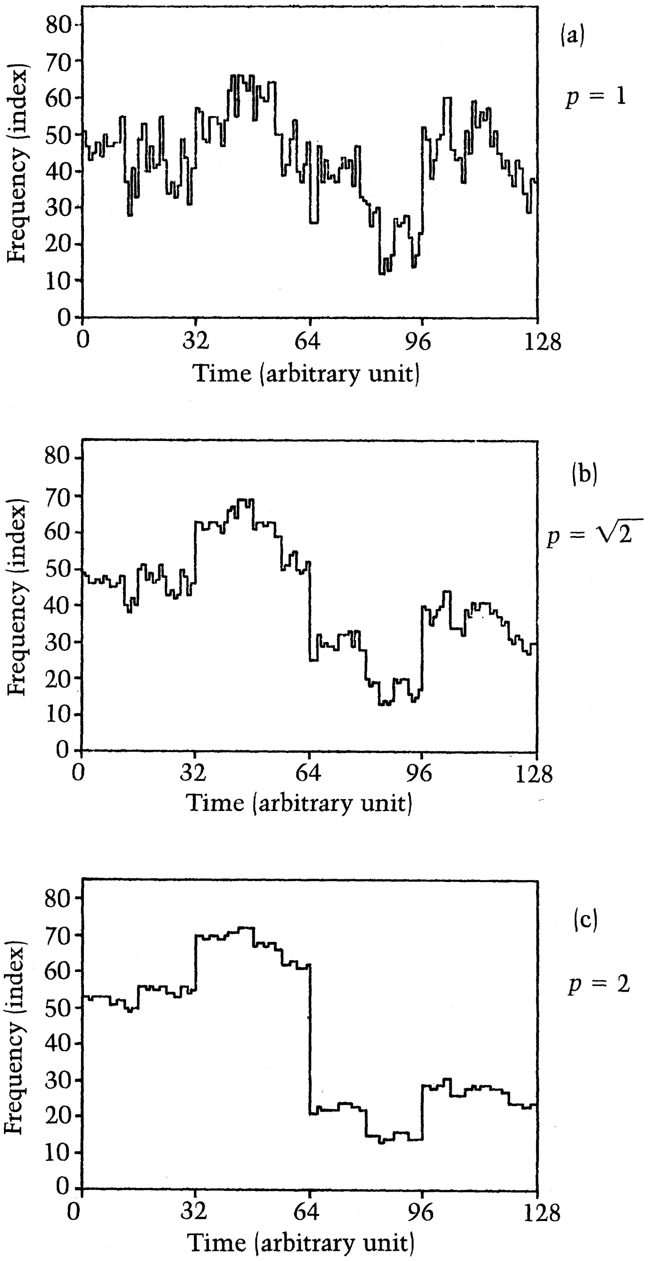
It is certainly not a coincidence that Voss's algorithm is also based on a set of time scales in geometric progression, $T_i = 2^i$, corresponding to the casting periods of each die. Moreover, it would not be too difficult to show that this set may be considered a deterministic and discrete equivalent of the continuous distribution of time scales introduced by Van der Ziel.

In conclusion, if a relation exists between $1/f$ spectra and hierarchical sets of time scales, and if we admit that such hierarchy may play an important role in musical structures, then we probably agree that the experimental observation of $1/f$ fluctuations in several musical styles is no longer so surprising.

Generating Melodies with Loaded Dice

We now extend the class of $1/f$ computer-generated melodies by modifying Voss's algorithm in two ways. As a first modification, the number of dice to be simultaneously recast at each step may be ran-

Fig. 5. Three melodies generated by the weighted-dice algorithm with different values of parameter p.



domized. In order to have a proper randomization of the casting pattern, the probability $p(m)$ of casting exactly the first m dice must be

$$p(m) = \begin{cases} 2^{-m} & \text{when } 1 \leq m \leq n-1 \\ 2^{-m+1} & \text{when } m = n \end{cases}$$

In this way, the average lifetime of the outcomes of a given die will be equal to the constant lifetime (casting period) of that die in the original algorithm. The purpose of this randomization is to preserve a hierarchy of time scales and, at the same time, to break its strictly binary organization and to let phrases be potentially divided into any number of subphrases.

In the second modification, every die is given a different weight. Dice are cast and recast in the usual (original or randomized) way, but the output numbers are now given by:

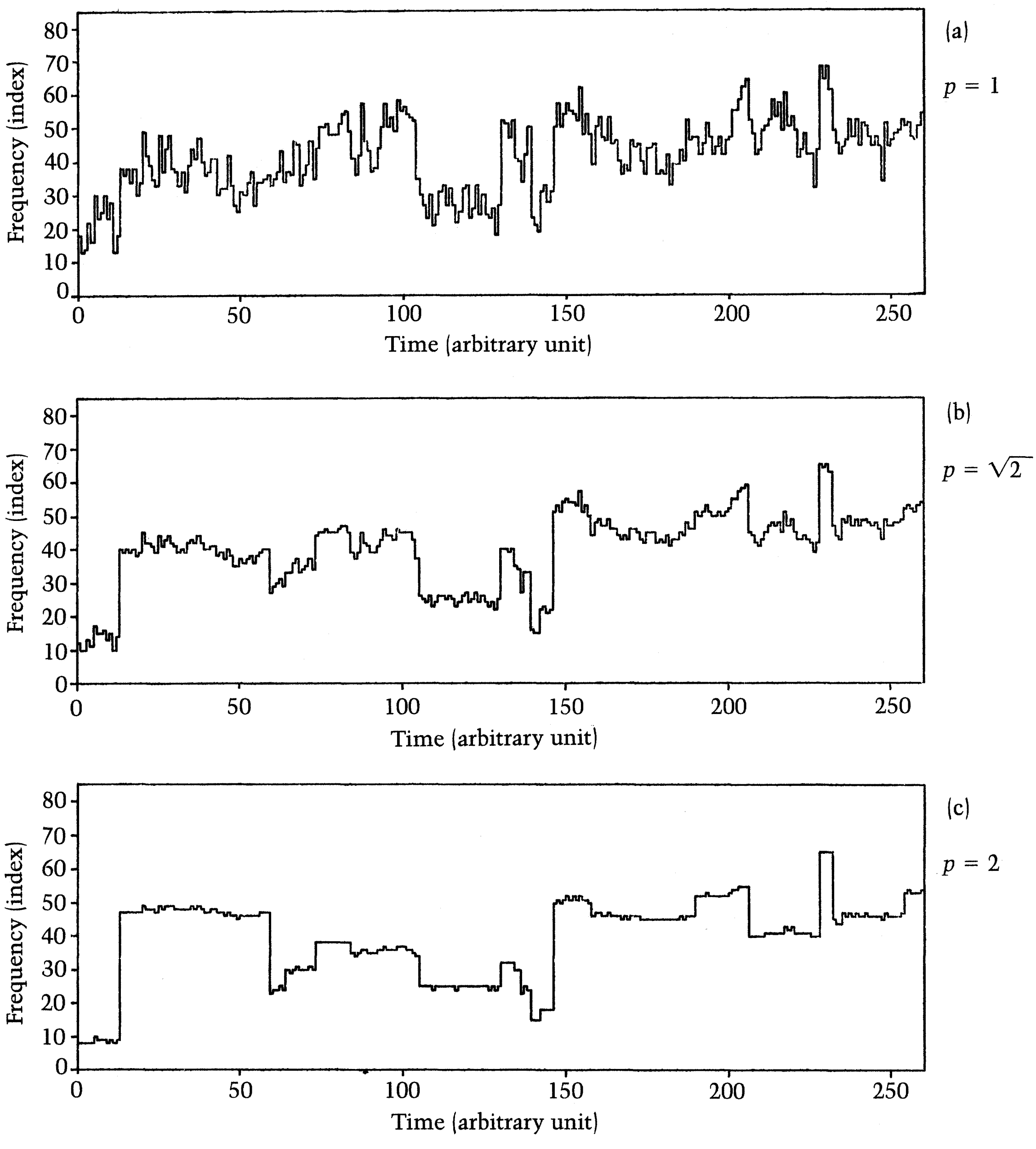
$$sum = \sum_{i=0}^{n-1} p^i die[i],$$

where $die[i]$ still represents the current outcome of die i, p^i is the weight associated with it, and p is a real number ($p > 0$).

Figures 5 and 6 show the effects of these modifications. A program simulates the casting of seven 13-sided dice and performs the weighted sums. The sums may assume 85 different values. (This number was chosen because the width of the frequency range of TAU2, the audio terminal used for these experiments, is exactly 85 semitones.) Obviously, the number and shape of the dice are arbitrary parameters.

Figure 5 shows three melodic lines of 128 notes each, generated by the weighted-dice algorithm without randomization. Die 0 has been cast 128 times, die 1 has been cast 64 times, and so on, until die 6, which has been cast 2 times only. In Fig. 5a, we have the special case $p = 1$, which gives the original algorithm. As p grows (Figs. 5b and 5c), the relative separation between adjacent subphrases also increases, and the binary hierarchical structure appears more clearly. The same seed for the generation of random numbers has been used in all three plots, to make them look like variations of the same structure. The three melodic lines in Fig. 6, with 256 notes each, have been obtained with the randomized version of the algorithm and with weighted dice as well. Again, the same seed has been used, and again it yields a hierarchical structure. As p is increased, it is clear that the hierarchical structure is no longer binary.

Fig. 6. Three melodies generated by the weighted-dice algorithm with different values of parameter p and with randomized die casting.



Self-Similar Music

The discussion about $1/f$ spectra and music leads us to the concept of self-similarity. There are at least two reasons why, in this section, we make self-similarity a new starting point for our experiments. First, the spaces of $1/f$ music and self-similar music overlap but are not identical. All three melodies in Fig. 5, for example, are self-similar, but only the first one has the $1/f$ property. This is because when the dice are loaded differently, the probability spectra change. Second, self-similarity probably speaks more to our intuition and imagination than do $1/f$ spectra. Therefore, we now set aside the $1/f$ requirement, as we already began to do with the loaded dice, and begin the exploration of the space of self-similar music by describing a new stochastic process.

Lévy Flight

A *Lévy flight* in an n-dimensional space is an infinite succession of isotropic and statistically independent jumps performed by a mobile point (to be called L) satisfying the distribution

$$P_r(x) \equiv \text{Prob}\,(r > x) = \begin{cases} x^{-D} & \text{if } x > 1 \quad (D > 0, \text{ real}) \\ 1 & \text{if } 0 \leq x \leq 1 \end{cases}, \tag{1}$$

where r is the random variable representing the length of the jump. A *Lévy universe* is defined as the set of points visited by L in its (Lévy) flight. Mandelbrot (1977) has shown that a Lévy universe is a set of points that are statistically (and asymptotically) self-similar. As Mandelbrot has also pointed out, its hierarchical structure makes it a suitable model for the distribution of stellar matter in the universe, with galaxies, clusters of galaxies, clusters of clusters, and so on. The exponent D in distribution (Eq. 1) measures the *fractal dimension* of the model universe and controls the spatial distribution of points (stars). As D decreases, clusters become relatively more separated, and their hierarchical structure appears more clearly. In this case, we say that the "degree of clustering" has increased.

If we apply a Lévy flight to music generation, we may control the degree of clustering of our hierarchical structures by means of a single parameter, D, in a way that may recall the weighted-dice algorithm explained earlier, where hierarchy was controlled by parameter p (Figs. 5 and 6). A difference between the two approaches is that the weighted-dice algorithm needs to keep *all* current values of the dice, while the only current value to be kept during a Lévy flight is the position of L, the mobile point. A more significant difference lies in the nature of hierarchy in the two cases. With the weighted dice, we have a discrete set of hierarchical levels, as many as there are dice, while the hierarchical structure of a Lévy universe, being uniquely determined by distribution (Eq. 1), appears as a continuum of levels.

Our experiments consisted of simulating Lévy flights within acoustical spaces. For this purpose, an *acoustic space* was simply defined as an n-dimensional space where each axis is associated with one of the acoustic parameters of sound controlled by the TAU2 audio terminal. Every point of the space corresponds to an array $(p_1, p_2 \ldots p_n)$ of values of acoustic parameters and may therefore be called a *sound event*. Performing a Lévy flight in an acoustic space means choosing an initial sound event and proceeding step by step, making L jump according to distribution (Eq. 1). Each jump extends the sequence by one sound event. Furthermore, as acoustic parameters vary within limited ranges, the acoustic space is also limited. To prevent the mobile point L from jumping out of the space, reflecting walls are used.

Figure 7 shows three monodic sequences of 500 notes each, generated by Lévy flights with different values of D placed in a one-dimensional acoustic space. The line indicates frequency. (The frequency range here is the full range of the audio terminal TAU2, which includes 255 values at a unit distance of one-sixth tone, corresponding to the already-mentioned 85 semitones.) Duration, timbre, and loudness, the other acoustic parameters controlled by TAU2, may be given arbitrary values.

As in the previous examples, melodic lines (in Fig. 7) seem to consist of phrases and subphrases with several hierarchical levels, reflecting the distribution of jumps in Lévy flights. Furthermore, the degree of clustering (in the vertical sense) of subphrases increases as D decreases.

Polyphonic Lévy-Flight Music

Musical applications of Lévy flight become more interesting when the process takes place in multidimensional acoustic spaces. In this case, we need to be more specific about the metric structure of

Fig. 7. Three melodies generated by Lévy flights with different values of the exponent D.

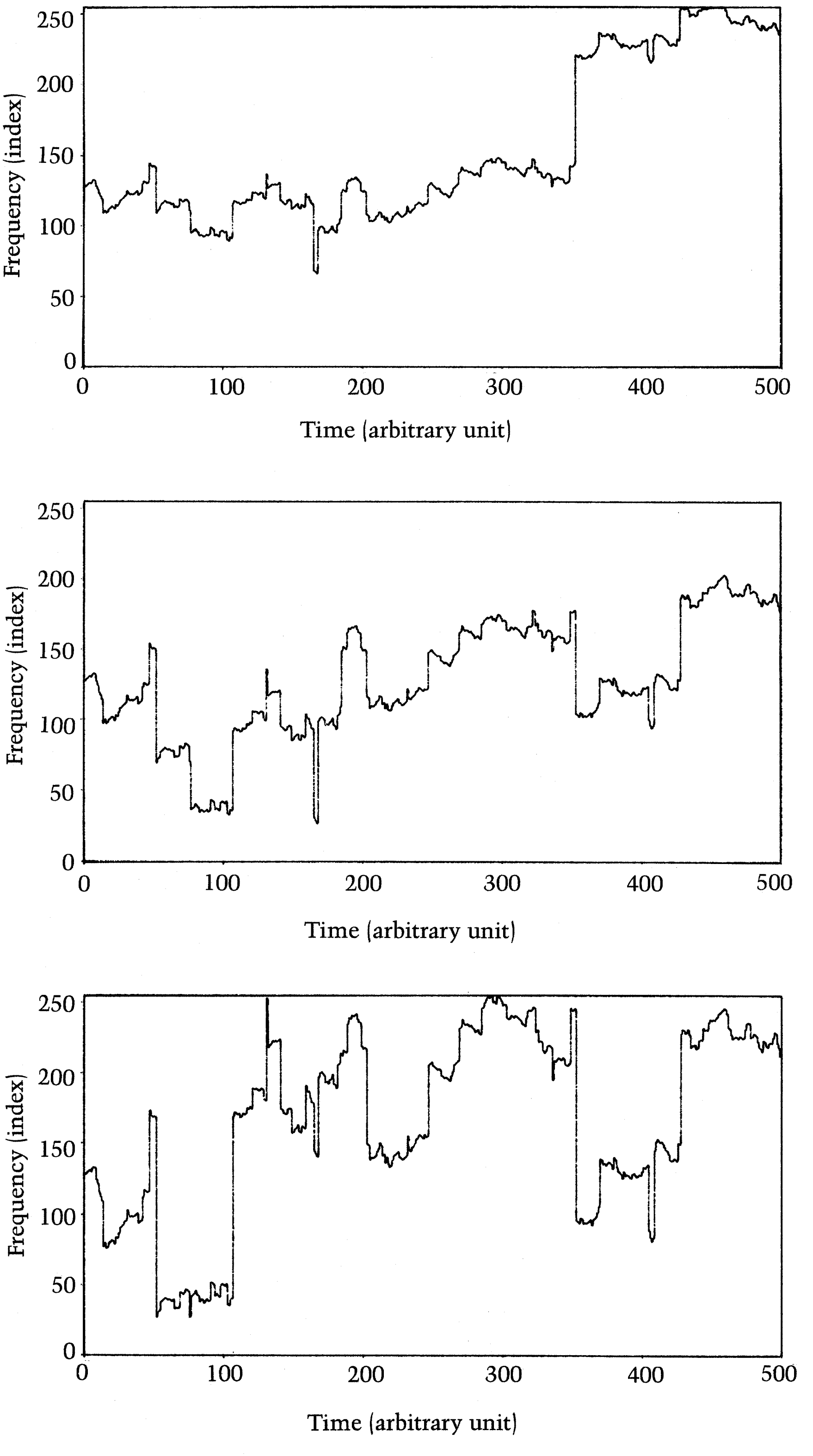
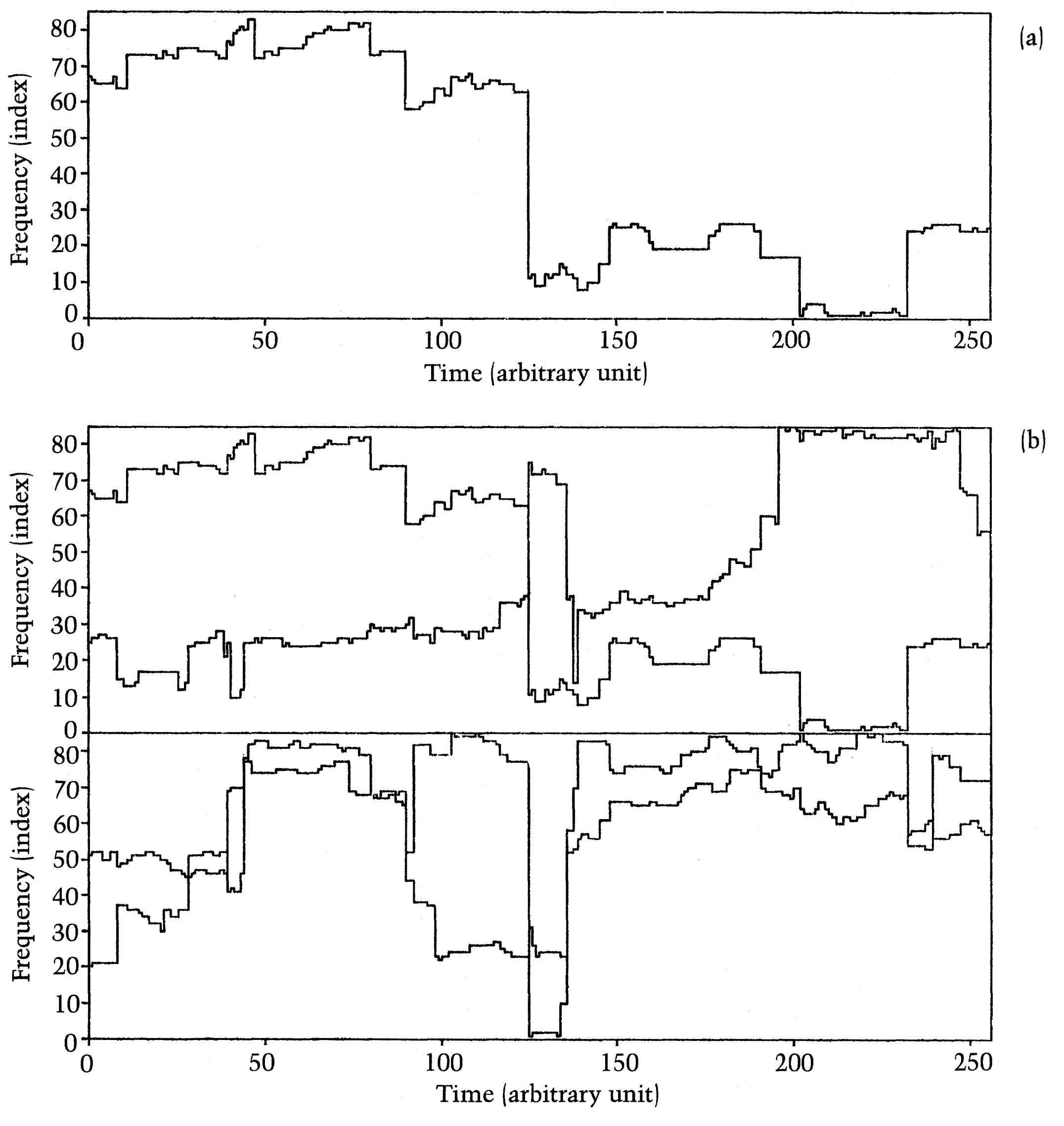


Fig. 8. A four-voice sequence generated by a Lévy flight (D = 1) in a four-dimensional acoustic space, where all four axes are associated with frequency.



the space, that is, the way distances between sound events are measured. In the examples of Fig. 7 we have implicitly assumed that the distance between two pitches of indices f_1 and f_2 (in the range [1, 255]) is given by $|f_1 - f_2|$ (but a quite different "style" of computer-generated melody was obtained by simply defining a metric where the fourth and fifth intervals are considered very short distances).

In the n-dimensional case, first we define a distance function d_i ($i = 1, 2 \ldots n$) for each acoustic parameter (axis) of the space. This function measures the distance between any two values of that parameter. Then we combine such functions to define the distance d between two sound events of the space. The distance $d(P, Q)$ between sound events $P(p_1, p_2 \ldots p_n)$ and $Q(q_1, q_2 \ldots q_n)$ may be defined in the following way:

$$d(P,Q) = \sum_{i=1}^{n} d_i(p_i, q_i). \qquad (2)$$

Fig. 9. A two-voice sequence generated by a Lévy flight (D = 0.8) in a four-dimensional acoustic space, where two axes are associated with frequency and two axes with note duration.

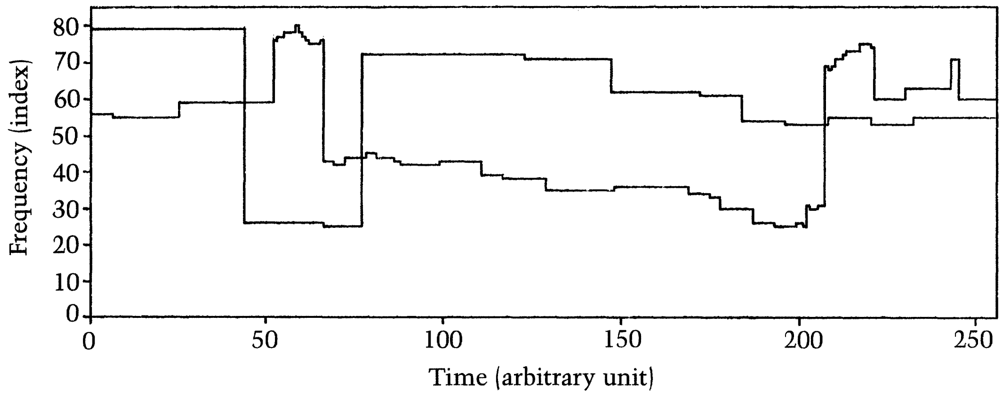
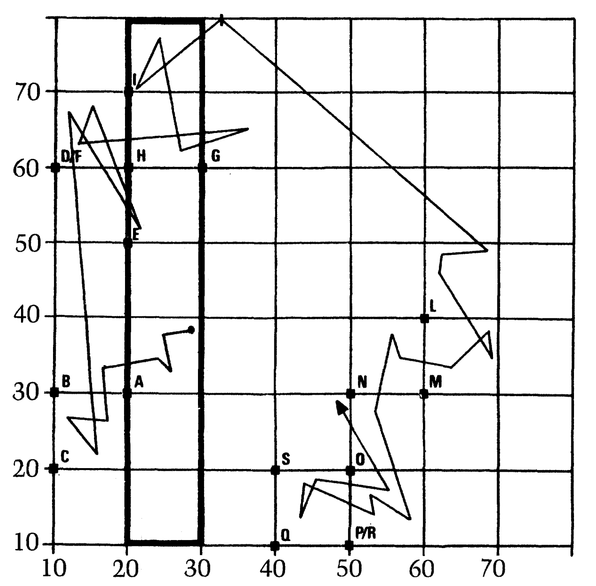


Fig. 10. A Lévy flight in a two-dimensional acoustic space covered with a grid and with reflecting walls.



The musical structures so obtained, which may also be polyphonic when frequency is associated with more than one axis, show an interesting property. When a mobile point L performs a Lévy flight with exponent D in the acoustic space, the n projections of L on the axes happen to perform n Lévy flights with the same exponent D. In other words, the self-similarity of the sequence of sound events is reflected in the fluctuations of each acoustic parameter and with the same degree of clustering as well. Furthermore, these fluctuations are interdependent, since jumps performed by each acoustic parameter are dependent random variables, being projections on different axes of an identical jump in the space.

Figure 8 shows a four-voice polyphonic sample generated by a Lévy flight in a four-dimensional acoustic space; all four axes have been associated with the frequency parameter. (Duration, timbre, and loudness of sound events are constant.) The frequency range includes 85 values. Figure 8a shows one of the four voices; in Fig. 8b, voices are compared two by two. It is evident how voices depend on each other: long jumps in different voices are often (but not always) performed at the same time.

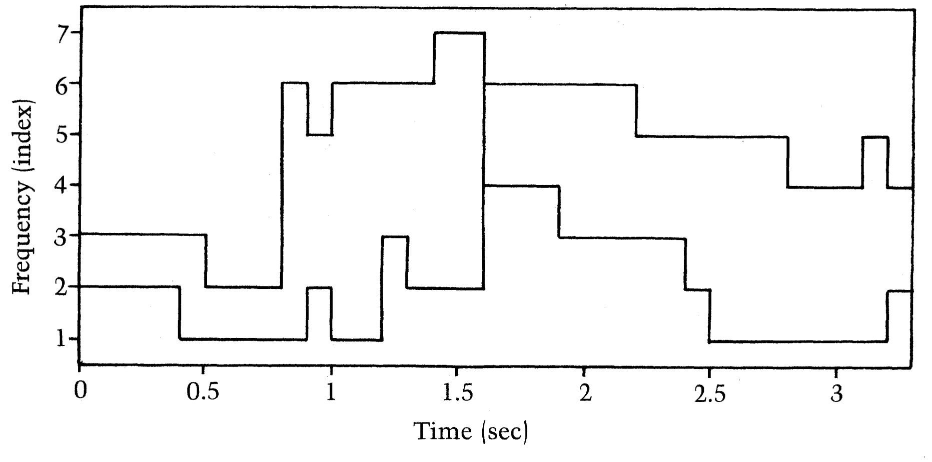
Figure 9 shows a two-voice sample generated by a Lévy flight in a four-dimensional acoustic space, where two axes have been associated with frequency and two axes with note duration. Notice that voices soon move as if they were independent of each other. Due to notes of different duration the mutual phase between the two voices is constantly changing, and frequency jumps corresponding to the same jump in the space are no longer simultaneous.

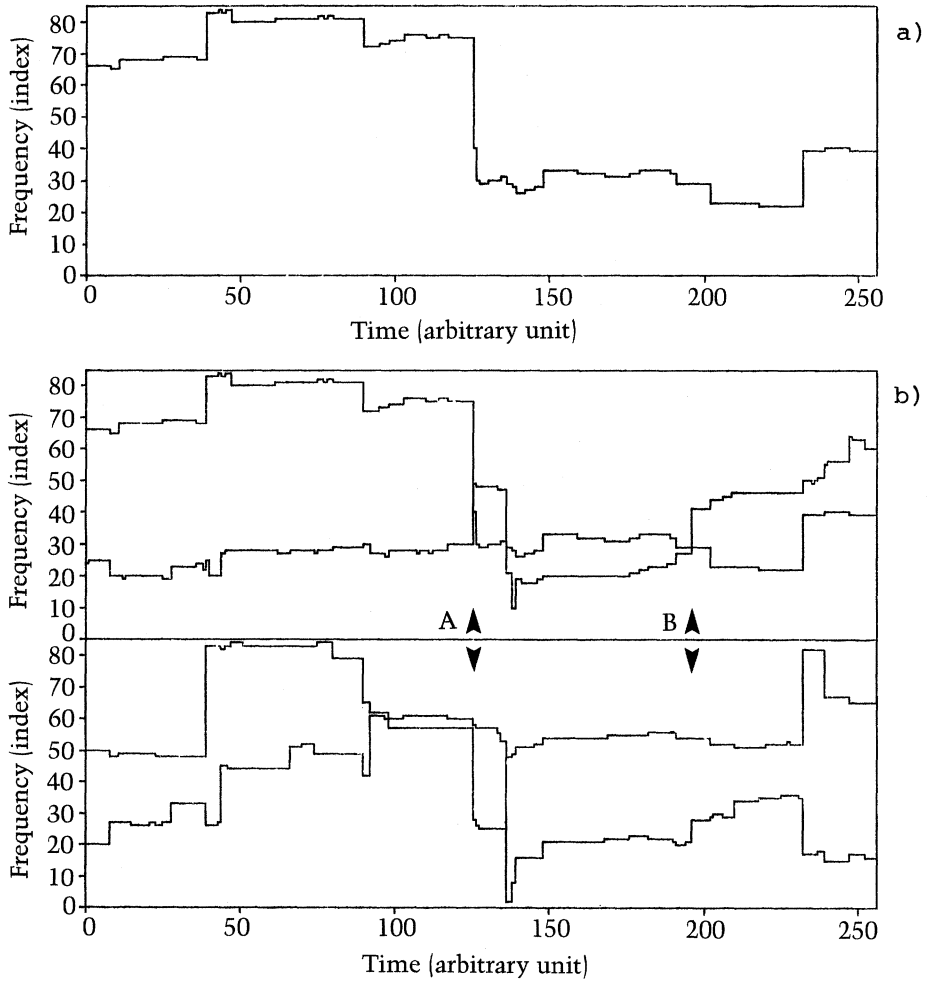
In the last application of Lévy flight discussed here voices are still synchronized, but only to a limited extent, in order to prevent the trivial chord-sequence effect.

Let a mobile point L perform a Lévy flight on a plane where an infinite square grid has been drawn. The set of squares visited by L during its flight, or rather the set of their vertices (for example, at bot-

Fig. 11. The two-voice sequence from the data of Fig. 10.

Fig. 12. A four-voice sequence generated by a Lévy flight (D = 0.7) in a four-dimensional acoustic space where all four axes are associated with frequency. (Note that durations are generated as in the example of Figs. 10 and 11.)





tom left in Fig. 10) is also a statistically and asymptotically self-similar set of points with the same degree of clustering of the set of the original stopovers.

Let us now consider a simple example. A square S of side $l_1 = 70$ length units is divided into $7^2 = 49$ small squares of side $l_2 = 10$. The sides of S, our acoustic space, are perfectly reflecting, and a mobile point L performs a Lévy flight in it (see Fig. 10). If we take the coordinates of the bottom-left vertices of the squares successively occupied by L during its flight (A, B, C . . . S), scale them by a factor of $1/l_2 = 1/10$, and consider the numbers obtained as indices in a frequency range (of seven values), we obtain a pair of dependent melodies.

Moreover, the grid implicitly provides notes with duration values, without affecting the mutual phase of voices, as happened in the previous example. Let L jump every 10^{-1} sec. Every frequency of the voice generated by abscissa (or ordinate) values may be assigned a duration equal or proportional to the time spent by L in the corresponding rectangle of dimension 10-by-70 (or 70-by-10 respectively). Thus, the first frequency index of the abscissa (x) voice will be $f_1 = 20/l_2 = 2$, because A's abscissa is 20, and its duration will be 4.10^{-1} sec. This is because L has occupied four points within the rectangle between abscissas 20 and 30 (the bold rectangle in Fig. 10) before jumping out of it.

Figure 11 shows the short Lévy duet obtained by the data of Fig. 10. As it appears from the figure, jumps of L may affect both voices, only one of them, or neither. Voices may appear locally independent, but still they are globally dependent and synchronized. Figure 12 shows a four-voice sample generated by a Lévy flight ($D = 0.7$) in a four-dimensional cube divided into cubes with side $l_2 = 10$. At time **A**, for instance, all voices move together, and three of them perform a relatively long jump; at time **B** only two voices move, while the others keep still.

Conclusions

Two sequences generated by the dice algorithm are included in the third album, *Computer Music*, published by the CNUCE – C.N.R. Institute at Pisa (Grossi et al. 1979). [Reviewed in *Computer Music Journal* 3(4): 58–59, 1979 — Ed.]

As an obvious extension of this research, timbre generation should be considered. Every self-similar object incorporates in its hierarchy both macro- and microstructures. It is therefore conceivable that a single self-similar stochastic process could generate both the overall structure and the sound texture of a piece.

References

Bertini, G. et al. 1980. "TAU2-TAUMUS: Il sistema di computer music in tempo reale realizzato a Pisa." *Automazione e Strumentazione* 28(2): 134–143.

Gardner, M. 1978. "White and Brown Music, Fractal Curves and 1/f Fluctuations." *Scientific American* 238(4): 16–31.

Grossi, P. et al. 1979. *Computer Music* (double album recording). Pisa: CNUCE – C.N.R.

Lidov, D., and J. Gabura. 1973. "A Melody Writing Algorithm Using a Formal Language Model." *Computer Studies in the Humanities and Verbal Behavior* 4(3–4): 138–148.

Lorrain, D. 1980. "A Panoply of Stochastic 'Cannons.'" *Computer Music Journal* 4(1): 53–81.

Mandelbrot, B. 1977. *Fractals: Form, Chance and Dimension*. San Francisco: W. H. Freeman.

Myhill, J. 1979. "Controlled Indeterminacy: A First Step Towards a Semi-Stochastic Music Language." *Computer Music Journal* 3(3): 12–15.

Van der Ziel, A. 1950. "On the Noise Spectra of Semiconductor-Noise and of Flicker Effect." *Physica* 16: 359–372.

Voss, R. F., and J. Clarke. 1978. "'1/f Noise' in Music: Music from 1/f Noise." *Journal of the Acoustical Society of America* 63: 258–263.