Composing Noise

Jae Ho Chang

June 1999

Institute of Sonology
Royal Conservatory
The Hague

Table of Contents

Preface 4

1 Introduction : A Brief History of Noise 5

1.1 The Definition of Noise 5
1.2 Non-Electronic Noises 8
1.3 Musique Concrète 12
1.4 Cologne Studio 13

2 Xenakis' Noise 16

2.1 Dynamic Stochastic Synthesis 17

3 Stockhausen's Noise 22

3.1 Material and Form 22
3.2 Compositions 25

4 Koenig's Noise 28

4.1 Funktionen 29
4.2 SSP 32

5 My Noise 35

5.1 Noise Interpolation Synthesis 36
5.2 Improvisation for Noise Solo 42
5.3 Miscellaneous Techniques 44

Conclusion 49

Appendix : **A. About Composing Noise CD** 50
B. Sound Examples 52
C. Compositions 54
D. Software 55

References 56

Preface

In spring 1996, I realized how beautiful noise can be. At that time, I had become interested in Chaos theory and begun to make ‘chaotic sound synthesis projects’. Most chaotic functions generate interesting visual patterns, but as sound their output values over time seem very random. When those values are used as sample values of a sound, the result actually sounds like white noise. To overcome this I reduced the frequency of the use of those values and filled the gaps with trigonometric interpolation. The result was high-pass filtered noise which was very pleasant to listen to.

However, the problem remained of the musical control over this noise. It was impossible to use musical parameters controlling the noise, because chaotic functions give discontinuous results for continuous changes in their parameters. Since then, I have become interested in other composers’ thought and techniques regarding noise, and developed my own.

This paper focuses on the noise synthesis techniques of three composers: Xenakis, Stockhausen, and Koenig, and it also discusses the noise synthesis techniques I have developed. Xenakis has radically and systematically used stochastic processes and probability theory. Stockhausen has used intuitive techniques with serialism as the fundamental principle. Koenig has developed and used techniques which combine aleatoric processes with serial techniques.

They all have different methods, but basically the same concept about noise. They did not distinguish dualistically between “musical sound” and noise. Instead, they thought of these as two endpoints of the sound continuum in which they have explored. This is fundamentally the same as the practice of composing new timbres instead of composing with instrumental or preexisting timbres. They wanted to be ‘composing noise’ rather than ‘composing with noise’. In other words, they wanted to create various sounds which often lie between musical tone and noise, rather than using white noise as a sound source.

Chapter 1 discusses the definition of noise and briefly shows the history of noise from the beginning of the 20th century to around 1950 when electronic music substantially emerged. Chapter 2, 3, and 4 contain the stories of Xenakis, Stockhausen, and Koenig, respectively. And the final Chapter 5 describes some projects of mine which are related to noise.

(Please see Appendix of this paper before starting to read. It might be useful for you to use “Composing Noise CD” which accompanies this paper.)

1 Introduction : A Brief History of Noise

> Wherever we are, what we hear is mostly noise. When we ignore it, it disturbs us. When we listen to it, we find it fascinating. The sound of a truck at fifty miles per hour. Static between the stations. Rain. We want to capture and control these sounds, to use them not as sound effects but as musical instruments.
>
> John Cage (1967)

Perhaps it was very long ago when people found potential power and beauty in noise. However, until the 20th century, noise was rejected as a musical element, or occasionally used just as a sound effect. It is the beginning of 20th century when composers began to pay attention to noise.

Some composers emphasized noise elements within conventional instruments, or used extremely dissonant chords in their work. There were also composers like Luigi Russolo who composed for innovative noise instruments which he made himself.

The availability of electronic devices from around 1950 gave composers more opportunities to work with noise. They could compose their own noises and they had better musical control over the synthesized or recorded noise. Composers like Pierre Schaeffer made music using recorded noises from the real world. Other composers, like Stockhausen, synthesized various noises using electronic devices such as the sine tone generators, noise generators, impulse generators and filters.

This history of noise from around 1900 till around 1950 will be described in a little more detail in this chapter. But first, the definition of noise will be discussed. Since the kinds of noises are unlimited and the esthetical approaches are various, there are too many possible definitions. Thus I will focus on the definition and aspects of noise in acoustical terms.

1.1 The Definition of Noise

In general, noise is often defined as any unwanted sound. Typically this could be various machine sounds from a construction area, the sound of a train passing near your house, loud Rock ‘n’ Roll music from the neighborhood, the sudden ring of a telephone at night. All of these sounds can be called “noise”, but this is unsatisfactory for our purpose.

The definition of noise in terms of acoustics is different from this general one. While the general definition is subjective and circumstantial, the acoustical one is objective and unconditional.

This chapter describes the definitions in three domains. The first and the second are the time domain and the frequency domain, respectively, which are two of the most important domains of acoustics. The third, the musical domain definition, describes how noise that has these acoustical definitions is used as musical material and how it is distinguished from other sounds by a composer. Since the role of noise in a music depends on the composer's subjective or aesthetic concept, we should look for the composer's own definition in his or her work beyond the acoustical definitions.

Time Domain Definition

In the time domain, noise can be defined as sound in which the amplitude over time changes at a certain degree of randomness. If the degree is maximum, the noise is completely aperiodic and becomes a 'perfect' noise (so-called *white noise*) in which the amplitude of any moment is not related to another in any meaningful way. Perhaps this 'perfect' noise does not exist, because most sounds have more or less correlation within themselves. According to the degree of this correlation or of randomness, the timbre or color of the noise varies.

The time domain is a very interesting one for noise synthesis, particularly for making an abstract model of noise synthesis. The techniques described in this paper have their own methods of generating noise in the time domain.

This domain, however, is disadvantageous in analysing sound; that is, it is very difficult to distinguish between noise and non-noise sound by looking at the time domain representation. There is a way to analyse sound in the time domain by drawing the *amplitude density distribution*. We can say that a sound is white noise if the amplitude probability distribution of the sound is Gaussian shape. But this is not enough for us to tell the characteristic of noise, because some kinds of noise such as "binary noise" has very different shape of amplitude probability distribution (Tempelaars 1996).

Frequency Domain Definition

In the frequency domain, noise can be defined as sound that has a continuous power density over a certain frequency range. When the power density of all the frequencies is equal, we call it *white noise*. The power density of pink noise decreases by 3dB per octave with increasing frequency, and that of brown noise decreases by 6dB per octave (Figure 1.1.1).

The frequency domain has the advantage in that it shows the characteristics of a noise rather clearly. When we see that the power density over a certain frequency range is continuous, we can recognize the sound has a noise element.

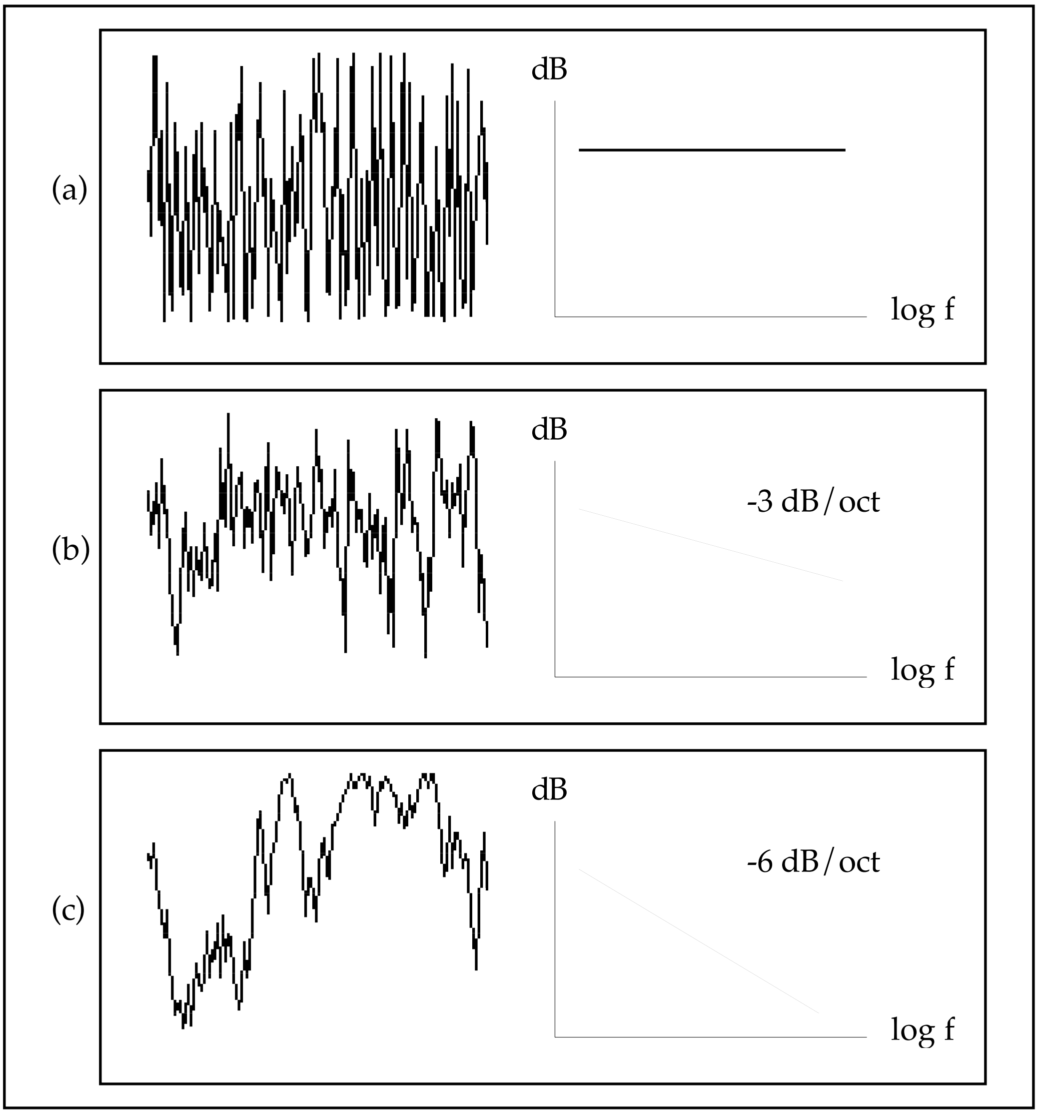

Figure 1.1.1 Noises in the time domain (left) and their density in the frequency domain (right). (a) White noise, (b) Pink noise, (c) Brown noise.

Musical Domain Definition

Noise has been redefined by composers according to their philosophy, frequently together with the basic acoustical definitions described above.

To Stockhausen, noise is defined as any kind of sound. A perfectly periodic sound, such as a sine wave, is thus the extreme case of noise (see Chapter 3). This might be the most reasonable (and general) definition in the musical domain because most sounds contain an aspect of noise as defined in acoustical terms.

To Xenakis, also, noise is one of the most important elements of music. He considered 'pure' electronic sounds produced by frequency generators as unnatural. He said that only when framed by other 'concrete' sounds, recorded from the real world, could electronic sounds could become powerful (see Chapter 2).

In Koenig's SSP (Sound Synthesis Program), there is no definition of noise because the composer does not design the spectral components of sound, but rather chooses the rules for directly generating sound sample values in the digital time domain (see Chapter 4). The composer, of course, can control the degree of noisiness by choosing various functions that have different degrees of randomness or repetition. The definition of noise in SSP, thus, depends on the composer's perception.

These definitions will be discussed more in detail from Chapter 2 on.

Conclusion

Noise components in a sound are often distinguished from periodic ones and are sometimes considered as non-musical elements. Much research has been done for noise reduction (Harris 1957). Periodic components of a sound are typically the focus of sound analysis techniques such as the Fourier transform and McAulay-Quatieri (MQ) analysis (McAulay and Quateri 1985). Sound synthesis techniques such as additive synthesis and the frequency modulation synthesis also focus on periodic components.

However, an interesting sound usually contains both periodic components and noise components, whose degrees change more or less over time. It has little meaning to claim that one sound is noise and that another is not. It is more accurate to state that one sound has a larger degree of noisiness and another has a smaller degree.

Therefore, sounds of all the degrees of randomness or correlation (or simply speaking, periodicity), including sine waves and white noise, can be used as musical materials.

1.2 Non-Electronic Noises

It was subsequent to the development of electronic music around 1950 that noise began to play an important role as a musical element in a musical work. However there have been many attempts to seek new timbres, musical noises among others, already from the beginning of the 20th century. Creating noise instruments, emphasizing the noise elements in conventional instruments, increasing the role of percussion instruments, and using extremely dissonant chords are some examples of that time.

The Art of Noises

In the early 20th century, a group of Italian composers, called Futurists, took various noises as a musical resource. Those noises were not new sounds to people, but rather sounds they were hearing in their daily life. It was, however, innovative to use those noises as musical sounds.

Luigi Russolo, one of the Futurist group, made a noise orchestra with various noise instruments, and composed some pieces using this orchestra. (The Futurist movement was primarily literary and visual. Russolo had joined this movement as a painter and later as a musician.) The instruments are named howler, roarer, crackler, rubber, hummer, gurgler, hisser, whistler, burster, croaker, and rustler. He wrote in his book *The Art of Noises* (first published in 1916; Russolo 1986) that his noise orchestra is the subsequence to some musical activities at that time : "In order to excite and stir our sensibility, music has been developing toward the most complicated polyphony and toward the greatest variety of instrumental timbres and colors. It has searched out the most complex successions of dissonant chords, which have prepared in a vague way for the creation of musical noise."

Unfortunately he did not describe how the instruments were built, but he wrote that each instrument had a certain range of playable pitch (usually 2 octaves). Those instruments, thus, were not just noise generators but real instruments.

His compositions were too innovative to be hailed by the audience. But this paradigm shift of him probably had some influence on contemporary composers.

Percussion Music

While Russolo made new instruments for musical noises, many composers have investigated new possibilities using conventional instruments.

We may say that percussion was the instrument group where the maximum degree of development and experiment has taken place. Percussion instruments that were minor in an orchestra and mostly used for effects began to be evaluated as musical ones. As the role of percussion has expanded, composers have used various sizes and types (metal, wood, cloth, glass) of beaters with conventional percussion instruments to experiment with new timbral possibilities. Exotic or foreign instruments have also begun to be used. Also, various playing techniques and new scoring has been developed.

The increase in the musical use of percussion eventually resulted in compositions for percussion ensemble such as Edgard Varèse's *Ionisation* (1931). In these works, what takes the major structural role (on a large or small scale) is not the melody or changes of pitch, as would be conventional, but the changes of timbre. This could be considered as the beginning of the techniques of 'composing noise'.

One particularly noticeable addition is the use of the brake drum. The fact that a part of a car can become a member of the percussion family shows the possibility that any object could be musically useful.

Sound Mass

There were composers who created noises using just the conventional techniques of the instruments. They have attempted to make musical noises by using strong dissonant chords.

Henry Cowell, in the 1920's, proposed building chords with major or minor seconds. He even proposed chords using closer intervals which can be derived from the 16th partial tone upwards of a naturally occuring fundamental. He used major and minor seconds to build chords and developed playing techniques such as pressing all the white or black keys, or both, in a certain range on the piano with the performer's palm or forearm. Such a chord is named by Cowell himself as a 'tone-cluster' (Cowell 1930).

Some composers created effects similar to tone clusters by using polytonality, or chords that have the characteristics of polytonality. Stravinsky, for example, in his work *Le Sacre du Printemps* (1913), made a fascinating noise for the harsh rhythm by adding together the two chords Eb7 and Fb (Figure 1.2.1).

These techniques largely succeeded in making musical noises by minimizing the pitch and harmony and at the same time maximizing the texture, timbre, and rhythm of a given passage. They have been continuously used by many composers. We can find examples in Luigi Nono's choral pieces like *Il canto sospeso* (1956), Penderecki's *Threnody for the Victims of Hiroshima* (1960), Górecki's *Sconti* (1960), György Ligeti's *Atmosphéres* (1961), and so on.

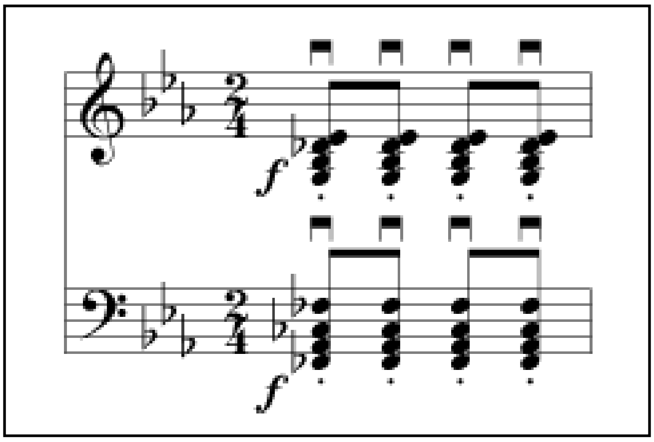

Figure 1.2.1 String section of a part in Stravinsky's *Le Sacre du Printemps*. Two chords Eb7 and Fb make a percussive noise for the harsh rhythm.

New Instrumental Techniques

We have already seen new techniques for percussion instruments. Many techniques for other instruments have been developed as well.

For string instruments, conventional techniques like plucking(pizzicato) possibly implied new possibilities. New techniques such as knocking the strings or body for percussive effect, bowing on the bridge or between the bridge and tailpiece, bowing with very strong pressure to make noise, playing or plucking the highest note, playing with irregular tremolo, have been created. Penderecki's *Threnody for the Victims of Hiroshima* (1960) is a good example.

For woodwind instruments, techniques like tapping the keys without blowing for percussive effect, multiphonics for new timbre, and so on, have been developed. The multiphonic technique is for producing two or more tones at the same time, forcing strong overtones by altering fingerings, lip pressure and position, and air pressure (Bartolozzi 1971). The sound we obtain with this technique is not a clear harmony, but rather quite a noisy timbre (Figure 1.2.2).

New techniques for brass, percussion, keyboard instruments, and voice have also been developed (Cope 1993). For keyboards, we can find the *prepared piano* technique which is for altering the piano timbre through the placement of objects such as clips, rubber, nuts, and stones on and between the strings, in compositions such as John Cage's *Sonatas and Interludes* (1946-48). For voice, noises such as laughing, talking and hissing have been added as musical expressions (for example, in Ligeti's *Aventures*(1962)).

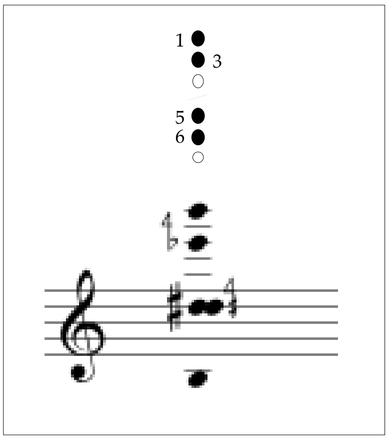

Figure 1.2.2 One of the multiphonics that the flute can produce. The upper part represents the fingering and the notes are what is actually sounding. This chord which contains microtones is not clear but makes a dissonant timbre.

Conclusion

The attempts described in this chapter had nothing to do with the techniques of electronic music. However, it is still interesting to explore yet newer instrumental timbres and compose with them if it can give us opportunities to make new and interesting musical forms.

1.3 Musique Concrète

Shortly before 1950, French composers Pierre Schaeffer and Pierre Henry began to use electronic media to compose music with various noises. While Russolo made the instruments that simulated noises from the real world, Schaeffer and Henry used these noises by electronically recording and manipulating them.

Early Recording Technology

It is only since the 1920s that musicians have used electronic recording technology. Composers such as Darius Milhaud, Paul Hindemith, and Ernst Toch experimented with variable-speed phonographs in concert (Ernst 1977). In John Cage's *Imaginary Landscape* No.1 (1939), the composer used two different-speed phonographs with records containing tones for testing purposes. However, such recording and reproduction systems were too expensive and too difficult to maintain and use until the end of the second world war.

It was only from around 1950 when the tape recorder began to be generally available and used. It was easy to use and less expensive. The tape recorder eventually made it possible for Schaeffer and Henry to realize, through electronic media, Russolo's concept of using everyday noises as musical elements, although it is not clear that they were influenced by Russolo.

Musique Concrète

In the 1940s, Pierre Schaeffer was working in a studio of Radio-diffusion-Television Francaise(RTF) in Paris as engineer and broadcaster. While working with recording devices in the studio, he realized its possibility as a compositional tool. After some experiments, in 1951, Schaeffer and Pierre Henry founded the Groupe de Recherches de Musique Concrète in Paris, and composed substantial works by using the *concrete* sounds recorded from the real world.

The sounds recorded on tape were used without any modification, or after being changed by tape manipulation and/or electronic modulation. The tape manipulation techniques included change of playback speed, reversed playback, editing (such as cutting and splicing), looping, and mixing. The electronic modulation was filtering. It was possible to make a sound that could range in timbre from the original sound to a sound which was radically and completely different from the original.

The recorded sound material was not synthesized but found or chosen. They were preexistent. This is similar to what French artist Marcel Duchamp did when he took various ordinary objects and constructed a work of art. On this point, Musique Concrète works are the opposite to the works of the composers who was working at the Cologne electronic studio. The Cologne composers used primarily synthesized sound, but not recorded sound from the real world (see Chapter 1.4).

Musique Concrète, however, soon gave influence to the other composers as well as the Cologne composers. Those influenced include Karlheinz Stockhausen (the major representative of Cologne composers), Olivier Messiaen, Edgard Varèse, Pierre Boulez, Luciano Berio, Luc Ferrari, Iannis Xenakis and so on. They have all worked in this studio in Paris. Many composers have used both recorded and synthesized sound for their compositions.

Conclusion

The recorded sound or noise was not able to play the role of a traditional instrument, because the manipulation of the possible musical parameters was too limited and not intuitive. Thus Musique Concrète did not perfectly accord with Russolo's thought and Cage's words quoted in a former chapter.

It is, however, highly significant that compositions were made via electronic media with the material of various recorded noises. The further developments in recording or sampling (the modern digital audio term) technology and various processing methods has given modern composers much greater possibilities. Musique Concrète is still alive.

1.4 Cologne Studio

The electronic studio founded in Cologne is potentially the most notable among the several studios built in various places in the world in the 1950s. Unlike the Musique Concrète composers, the composers in Cologne tried to use only the sounds generated with electronic devices for their compositions. After a few years, however, this idea was abandoned. In 1955-56, Stockhausen used recorded sung sounds together with electronically generated sounds in his work *Gesang der Jünglinge*.

The Studio

The Studio for Electronic Music of the Westdeutscher Rundfunk (West German Radio), Cologne was founded in 1951 under the direction of Herbert Eimert. At that time, there were basic electronic devices such as sine tone generators, white noise generators, filters, ring modulators, loudspeakers, and tape recorders. There was also some electronic musical instruments such as the melochord and trautonium, but there were rarely used

Contrary to the quickly presented results of the Musique Concrète composers, the efforts made in the Cologne studio were quite slow to bear fruit. There were small activities such as Radio broadcast and public demonstrations, but the first real concert took place in 1954, three years after the foundation of the studio (Eimert 1955). The reasons are not only that this kind of music was too new, but also that it required a very great effort to produce and thus a long time to work with the devices in the studio to achieve musical results.

Composers & Compositions

The composers who worked in Cologne in those early days are Eimert, G.M. Koenig, Karlheinz Stockhausen, Karel Goeyvaerts, P. Gredinger, Henri Pousseur, Ernst Krenek, György Ligeti, and so on. Works such as Eimert's *Fünf Stücke*, Koenig's *Klangfiguren*, Krenek's *Oratorio for Pentecost*, Stockhausen's *Gesang der Jünglinge* were composed at that time.

Every composer of this time may have had a different attitude to electronic music, but their attempts had at least the similarity of not trying to replace conventional instruments nor imitating instrumental music. Stockhausen and Koenig (described in Chapters 3 and 4) are no exceptions. They had new materials that were emancipated from the language of instrumental music.

Eimert classifies the sounds of the electronic music at the time as follows:

1. Tone

    Tone is a sinusoidal wave which does not have any overtones. This is different from that in instrumental music. Eimert calls the tone in instrumental music *note*. Although the terminology does not matter, the concept of the *tone* in electronic music does not exist in instrumental music. Any conventional instrument produces sound that has its own preformed overtones due to its physical structure. The concept of *tone* in electronic music means the breakdown of the limit of such preformed timbre.

2. Note

    Note is a sound that is built up from a series of harmonic overtones. As mentioned above, this is the same as the *tone* in instrumental music in terms of the definition. The conventional *tone*, however, is actually different from the electronic *note*. To imitate the expression of a conventional instrument was not possible and meaningful. But a composer can control each overtone in a note, and this makes it possible to express in electronic music something that is difficult in instrumental music, such as timbral transition.

3. Note mixture

    Note mixture is similar to the *note* but has non-harmonic partials. There could emerge some noise elements due to the non-harmonic relationship of the partials.

4. Noise

It does not seem amazing that the noise generator was used from the beginning of the studio. It might be natural that composers considered using *noise* which is spectrally rich, instead of using the superimposition of sine waves to make complex sounds. Most composers used noise together with a bandpass filter, and controlled the setting of the filter to get different pitches and timbres.

5. Chord or note complex

Chord or *note complex* is the same as the traditional *chord*. While note and chord in instrumental music are clearly distinguished, those in electronic music are on a continuum. As mentioned above, each overtone or partial can be controlled separately, and therefore a note mixture can be turned into a chord and vice versa.

6. Impulse or pulse

Impulse or *pulse* is also called *beat* or *click*. An impulse can produce noise-like sound which is explosive at high intensity. The famous use of impulse is found in Stockhausen's *Kontakte*. He recorded a few pulses on a tape in a simple rhythm, made a tape loop with it, and sped up the tape until it became a *note*. In this way he made a certain sound which has a 'composed' timbre.

There is no doubt that it was very difficult and took a great time to make these materials with the electronic devices of the time and to make a musically interesting composition with them. For example, Stockhausen described the creation of his work *Kontakte* as follows : "In some sections of Kontakte, I had to splice it all by hand, which is an unbelievable labour. I worked on the last section of Kontakte, ... together with Gottfried Michael Koenig, ... for three months. And when it was completely ready, I spliced it together with the previous sections, listened, turned pale, ... And I came back next morning and announced to Koenig that we had to do it all over again. (Stockhausen 1991)"

These difficulties at that time, of course, have shown the need for more and better devices. The emergence of synthesizers and computers has solved some of the problems of the time, and have furthermore become useful tools for the development of new techniques, new sounds, and new form of music.

2 Xenakis' Noise

> Instead of starting from the unit element concept and its tireless iteration and from the increasing irregular superposition of such iterated unit elements, we can start from a disorder concept and then introduce means that would increase or reduce it.
>
> Xenakis (1992)

It is to Iannis Xenakis that one should direct attention among the composers in the 1950s who worked with both the Musique Concrète and Cologne studio techniques. His early works such as *Diamorphoses*(1957) are affected by Musique Concrète, but most of his works are instrumental pieces made with probability theory (he called "Stochastic Music"). Later he also used the mathematical theories of Games ("Strategic Music"), and Sets and Logic ("Symbolic Music") (Xenakis 1992).

In the 1990s, Xenakis realized the application of probability theory to sound synthesis. He described the basic techniques in the first edition (1971) of his book *Formalized Music* and has developed *Dynamic Stochastic Synthesis* method which is described in the next chapter. His works such as *Gendy3*(1991) and *S709*(1994) and are composed with this method, *La Légende d'Eer*(1977 - 1978) uses elements synthesized with an earlier implementation of the concept.

Xenakis thought of noise as sound made by a particle that randomly moves along the amplitude ordinate. He explores the space between disorder and order, between musical tone and noise, by giving certain constraints to the particle.

This idea is constructed on the premise that sound synthesis and analysis based on the Fourier series has failed. Xenakis argues that the superimposition of trigonometric functions which consist of a unit element of variations within 2π and its infinite iteration, is too simple to create an interesting timbre because the ear easily detects the temporal pattern in the sound.

Instead, Xenakis made an abstract model in the time domain which has a disorderly wave into which one can introduce periodicities and symmetries. The greater the introduced periodic behaviour, the less like noise the sound becomes and vice versa. This model is described in more in detail in the next chapter.

2.1 Dynamic Stochastic Synthesis

Dynamic Stochastic Synthesis is an abstract and mathematical sound synthesis model proposed and developed by Xenakis in which probability theory is used. He wrote about the use of probability theory on the microlevel as follows : "The solutions in macrocomposition ... (programmed stochastic mechanisms) can engender simpler and more powerful new perspectives in the shaping of microsounds than the usual trigonometric (periodic) functions can. Therefore ... we can bypass the heavy harmonic analyses and syntheses and create sounds that have never before existed. (Xenakis 1992, Preface)"

The basic method of sound synthesis that Xenakis proposed is to directly control the movement and shape of the waveform with stochastic processes and variations. This is the same principle as having a particle randomly move along an ordinate. Xenakis proposed eight ways to control this random movement or random walk to make sound (Xenakis 1992, pp. 242-254). For the selection of random values, various distribution functions can be used (refer to [Lorrain 1980] for these functions) and the results are very different according to the chosen function.

Based on these methods, Xenakis describes Dynamic Stochastic Synthesis, which is systematic and musically useful, in the second edition of his book (Xenakis 1992).

The Algorithm and The Elements

To summarize Dynamic Stochastic Synthesis, it is the somewhat controlled variation over time of a waveform. In this algorithm, the composer has the role of choosing the random and restricting parameters for creating a waveform, setting some constraints and then letting it be evolved by itself. Of course for making music, the composer should choose parameters that create interesting sounds.

The*Waveform* consists of a number of segments given by the number of *endpoints* (Figure 2.1.1). The time and the amplitude values of each endpoint randomly walk by the given distribution function. The ranges of these random walks are limited by the given *barriers*.

These elements are summarized as follows:

1. Waveform

   The sound is the result of evolution of a waveform over time. A waveform consists of the given number of endpoints. The length of each waveform is determined by the movements of the endpoints on the time axis. That is, the exact pitch, the exact amplitude and the timbre of the sound are not directly specified by the composer but are determined by other elements selected by the composer and change continuously.

2. Endpoint

   Endpoints form a waveform. The samples between endpoints are calculated by linear interpolation (Figure 2.1.1). When a waveform is completed, each endpoint randomly walks along the time axis as well as the amplitude axis. The shape of the next waveform, thus, is determined by those movements (Figure 2.1.2).

3. Random Walk of the Endpoints

   The movement or random walk of each endpoint is determined by a given distribution function. Various functions can be used for a different shape of the movement.

4. Barriers

   Barriers confine the movements of the endpoints within specified ranges. They are reflecting, so that an endpoint that goes over one of these barriers is reflected by the amount of excess as in Figure 2.1.3. There are four barriers, which guard minimum waveform length, maximum waveform length, top amplitude value, and bottom amplitude value, respectively. These barriers also can randomly walk like the endpoints.

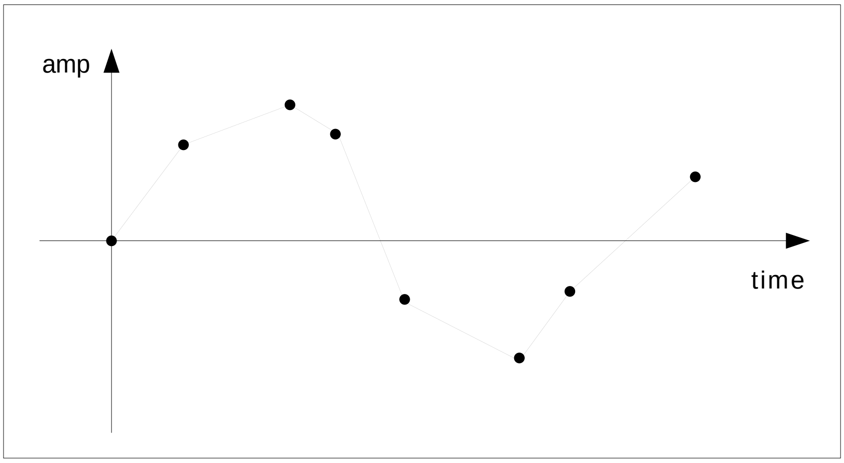

Figure 2.1.1 A waveform with 8 endpoints. Between the endpoints, the samples are calculated by linear interpolation.

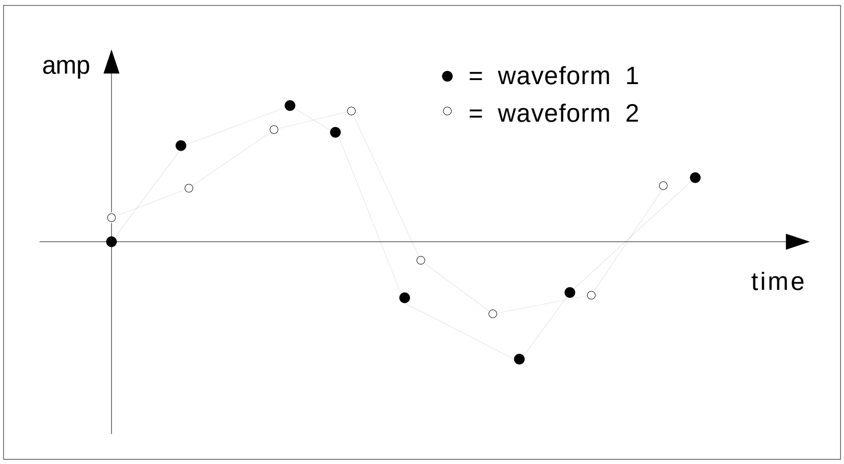

Figure 2.1.2 A waveform (waveform 1) and its first evolution (waveform 2).

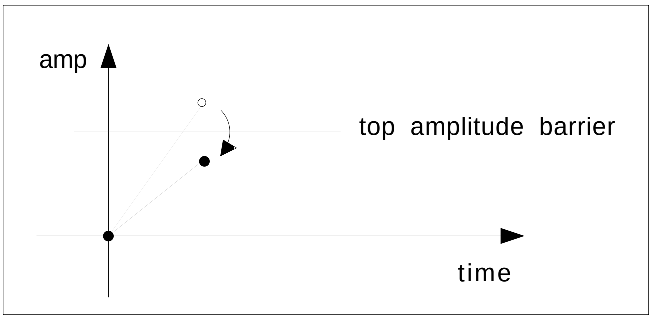

Figure 2.1.3 An endpoint is reflected when going over a barrier.

Implementation & Making Music

The first implementation was the computer program GENDY (Génération Dynamique) developed by Xenakis with assistance from Marie-Hélène Serra. This program was written in BASIC. Figure 2.1.4 shows its block diagram. In this diagram, only one sound or voice can be made, but Xenakis used 16 voices in his work *GENDY3*.

In the GENDY program Xenakis explores one of his philosophical desires, to create a piece of music from "nothing", simply by determining the environment and initial parameters for its creation. The composer determines the length of the whole piece, the division and order of sections, the number of voices which should be used in each section, the density and activity of each voice. The composer also specifies parameters for each voice, as described above. These parameters include the type of distribution functions, the position and movements of the barriers and so on. The composer chooses interesting setting through experimentation (Robindoré 1996), but the decision of weather a voice is on or off, whether there is sound or silence, is determined by the "Formal Architecture" (consisting of a Duration Random Walk and Bernoulli Trial) as depicted that in the upper part of Figure 2.1.4. This part determines for each voice the start time, end time and duration.

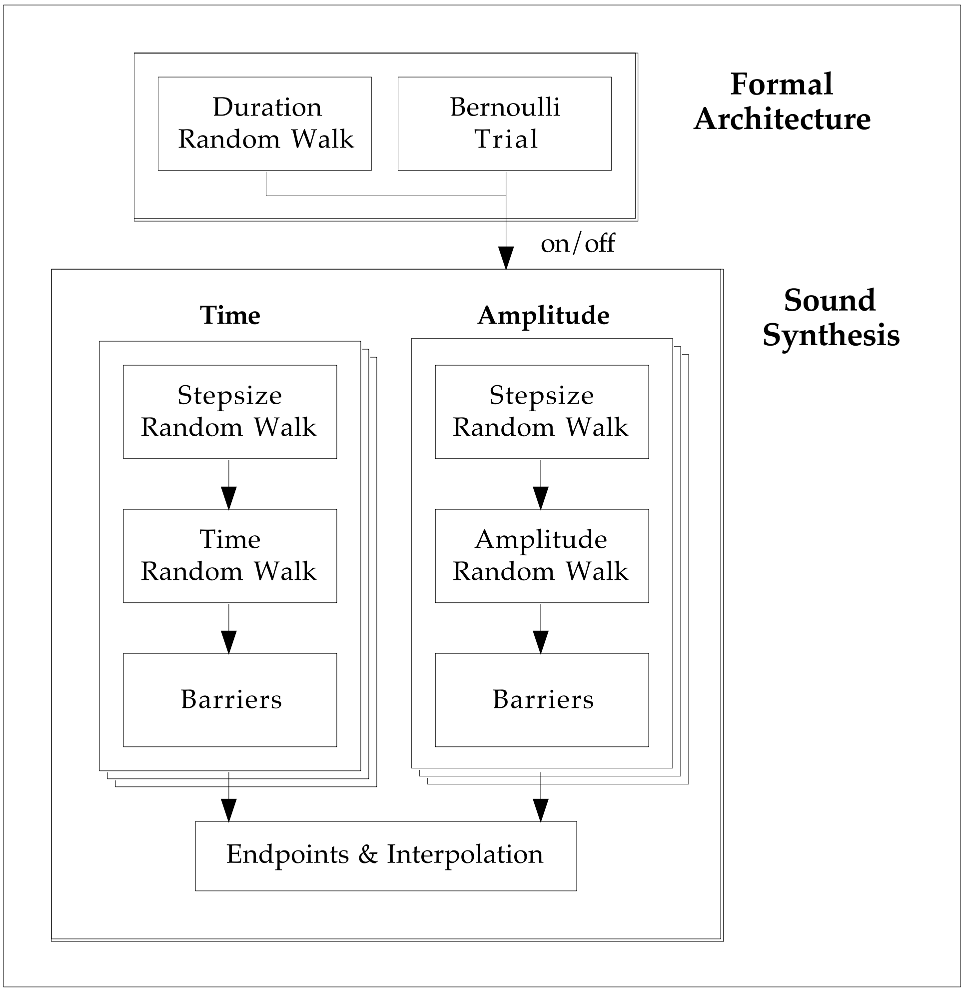

Figure 2.1.4 The block diagram of the GENDY program (one voice).

The GENDY program did not have a user-interface for parameter input. Xenakis wrote all the parameter values directly into the program text. And although this program makes 16bit digital sound, there is much quantization noise due to the limited resolution inside the program. (For example, the range of amplitude is -100 to 100 instead of -32767 to 32767. However, one could say that this sound quality is part of the attractiveness of *GENDY3*.)

Later, this program is reengineered by Peter Hoffmann. The new GENDYN program runs on a PC in real time and has a graphical and interactive user-interface (Hoffmann 1997).

Xenakis himself has modified the program by adding time-variant aspects, and with this variation he completed the work *S709*. This version has been used for this specific work and it has not been documented or released for general use. If this algorithm was implemented with the addition of a modern user interface to allow rapid interaction and user customization of the algorithm, it would be an interesting experimental sound synthesis platform. Many variations and extensions are possible. For example, the user could apply specific rules to each endpoint and more various distribution functions could be used for the random walks. Different interpolating functions other than linear one could be used, because it could radically influence the timbre. Different ways to control the overall form could be applied as well. There is scope for further development.

The Sound of Dynamic Stochastic Synthesis

Dynamic Stochastic Synthesis creates sound out of nothing. That is, all the endpoints of a waveform start at zero amplitude. Each endpoint goes its own way. All the musical elements such as pitch, timbre, and intensity, are the results of the movements of the endpoints.

The creator or composer of this sound, however, sets rules or constraints for the movement of an endpoint. Variations in these constraints create very different sounds. The important things in this algorithm are distribution functions that determine the characteristic of the random walks of the endpoints and the step size of the random walks (the composer does not directly set this, but it also randomly walks according to the distribution function chosen by composer.), and the range of the random walks.

The type of distribution function radically determines the characteristic of a random walk. For example, the hyperbolic cosine distribution makes a smooth random walk due to its tendency towards small values, the logistic distribution makes a coarser random walk because it generates larger values than the hyperbolic cosine, and the Cauchy distribution makes very unstable random walks with sudden changes because it generates both very small and very large values.

The range of movements on the time-axis influences the periodicity or randomness of the sound. For example, if the movement is very much restricted, the sound becomes almost periodic. If the movement on the time-axis is more unrestricted and the logistic distribution is applied to the movement on the amplitude-axis, the sound becomes very noisy.

This algorithm is not useful for making common instrumental timbres or preexisting sounds, with the very pitch we want at this moment. This algorithm is a tool for designing and creating the lives of waveforms to obtain unique timbres. Sounds created this way have no relation to physical processes and are not available from other sound production methods.

Conclusion

Probability theory is attractive because a stochastic system, which is based on it, like Dynamic Stochastic Synthesis seems aleatoric, random and unexpectable, but its life eventually shows its deterministic feature. This algorithm can also lead to sound that seems randomly moving to a deterministic state. This is an exciting and powerful paradigm for sound synthesis.

3.0 Stockhausen's Noise

> The category of noise is no less differentiated than the category of sounds. It is natural that in the new musical language, non-periodic phase-relationships determine all aspects of the form - on the small scale as well as on the large scale; periodicity thereby becomes an extreme case of non-periodicity.
>
> Stockhausen (1959)

Stockhausen has emphasized the importance of the continuum between periodic waves and noise. His continuum used in his works is, however, discrete unlike that of Xenakis shown in the previous chapters. For example, Stockhausen made a scale by arranging basic sound materials along the continuum between periodic waves and noise according to the degree of randomness in the sound material. We cannot find in Stockhausen's works the continuous change of the degree of randomness which can be found in Xenakis' Dynamic Stochastic Synthesis. This is because Stockhausen's approach is very different.

We can summarize Stockhausen's techniques of 'composing noise' as developing a scale of sound materials and serial manipulation over it. This is based more on his musical intuition rather than on a mathematical, acoustical or systematic model or algorithm. Perhaps this is not a musical tool that we can use as is, such as the techniques of Xenakis and Koenig. However, studying Stockhausen's methods may still open a path for us to use our own musical imagination.

3.1 Material and Form

Stockhausen wrote about the relationship between material and form (Stockhausen 1959). His thought could be summarized as follows: A new form requires new materials. And new materials construct a new form. A new architectural form has required concrete and aluminum instead of wood and bricks, and at the same time, such new materials has given birth to new architectural forms. A new music also requires new instruments and new instruments construct new music.

To Stockhausen, the best way forward is to start disassembling the sounds of conventional instruments to obtain new musical materials. He draws the continuum of periodic waves and noise from this.

Conventional Instruments : Ready-made Objects

Conventional instruments are too limited to make new forms of music. We cannot easily have them play a scale which consists of 13 or 14 tones in an octave. We cannot write a passage in which a clarinet sound gradually turns into piano sound. We cannot order a flutist to play a tone for 10 minutes without taking a breath.

Traditional instruments are like ready-made objects. Composers do not build violins or pianos themselves. They just use them. Composers must fit their music to the fixed ways of the instruments.

New music, of course, has been developed continuously through the experimental methods that we have seen in Chapter 1.2. In addition to those methods, many compositional experiments have made new music. There have been pointillistic pieces composed by the enthusiasts of Anton Webern and integral serialism by Luigi Nono, Pierre Boulez and so on. Composers such as Morton Feldman and John Cage made indeterminate and aleatoric music. (Stockhausen also was one of those composers. He pointed out the limitation of conventional instruments, but he did not ignore them. Rather he wrote more works for conventional instruments than electronic music.)

These developments, however, could not completely open the whole world of music. The ready-made instruments eventually had to be abandoned for new forms of music. It was therefore natural that electronic devices became progressively more interesting to composers.

Continuum between Sine Wave and White Noise

Electronic devices that were new musical instruments offered opportunities for Stockhausen to build new musical forms. Stockhausen's basic materials were sine waves and white noise. He says that sine wave can be considered as a *vowel* sound and noise as a *consonant* sound.

The sine wave generator provides an instrument with which composers determine and control each component in a spectrum instead of the fixed spectrum resulted from the fixed structure of a conventional instrument. To compose with this new instrument, however, was very difficult and took a great deal of time. Typically, a composer records a sine wave on a tape, adds second one, a third one and so on. The sound is therefore the result of a compositional act.

The white noise generator together with filters makes various kinds of 'consonant' sounds. The consonant is indispensable element in most spoken languages, because we cannot communicate each other without it. Noise plays a similar role in conventional instruments. We know it through this experiment: when we record an instrument sound and cut the first part that contains the irregular and unstable startup of the vibration, we cannot recognize any more which instrument it is. This is a physical phenomenon. This noise, however, now became an important musical material.

Stockhausen emphasizes the continuum between sine wave and noise. Filtered noise(so-called *colored noise*) has a certain width of frequency band. When this bandwidth is the widest, the sound becomes white noise, and the narrower the bandwidth, the sound becomes more periodic. If the bandwidth can be reduced to 1Hz, we would obtain a sine wave. In other words, a sine wave is the narrowest bandwidth noise, and white noise is the densest superimposition of sine waves. This means that sine wave and noise are connected to each other and two extremes of the same continuum. This is definitely an innovative musical concept.

Intuitive Music

Stockhausen, however, said that instrumental music should be related to the free and intuitive sense of human beings. Stockhausen used indeterminacy and aleatory techniques in his compositions, as well as intuitive techniques (Stockhausen 1991).

In *Zeitmasse*(1955-56), indeterminacy of tempo is used. For example, all of the instruments begin together but each with a different tempo. One player should play their part as quickly as possible, another as slowly as possible, another slowly then faster and faster and the others at fixed tempo.

In *Klavierstück XI*(1956), indeterminacy of form is used. The score consists of 19 long and short fragments dispersed on it, the player randomly chooses one of them and plays.

In *Mixtur*(1964), there are sections that have indeterminate pitch or indeterminate rhythm, or both. These sections contain sound masses.

In *Aus Den Sieben Tagen*(1968), there is even no musical material. Instruments or the number of player are not given. There are only 15 texts and some simple instructions. The players, whatever their instruments, improvise intuitively according to the given texts.

Stockhausen realized such indeterminate and aleatoric forms with conventional instruments, and supposed that it would be hard to realize them with electronic devices. He said : "In contrast to electronic music, where all sound events are predetermined down to the smallest detail and fixed by technical measures, the player of this new instrumental music is granted opportunities for free, spontaneous decisions inaccessible to machines. (Stockhausen 1959)" This would appear to be only a technical problem of that time. For example, it was practically difficult to realize a composition in which a few players improvise with a number of sine wave generators. However, this can be easily achieved with the electronic devices of today.

I think that such technical difficulty is not the only reason of that Stockhausen has used conventional instruments for the compositions mentioned above. It is possible that he was also attracted to the language of conventional music.

Conclusion

The possibilities of synthesis techniques for noise seems to have offered another compositional dimension to Stockhausen. This is the scale of timbre which will be described in the next chapter. This allows the composer to change the traditional attitude that dualistically distinguishes between periodic waves and noise, or instrumental tones and percussion sounds and makes a new form on the continuum (discrete in Stockhausen's case) between those two.

3.2 Compositions

In this chapter, Stockhausen's techniques of noise composition are examined by taking his works *Gesang der Jünglinge* and *Kontakte* as examples. As mentioned before, the fundamental composition technique is based on serialism.

Gesang der Jünglinge

In this work(1955-56), he used both Musique Concrète and electronic sounds. Fragments of a boy's sung voice were recorded on a tape and the various techniques of Musique Concrète were applied to the tape. The electronic sounds were made with a similar process using the same equipment used in *Elektronische Studie I* (1953) and *Elektronische Studie II* (1954) (with one exception that statistical control methods over the electronic devices were used(Stockhausen 1991, pp. 45-46)).

Stockhausen wanted to unify these two kinds of sound. That is, his intention was not the comparison or contrast between the two sounds. To achieve this, he arranged all the sounds, whether it is recorded or synthesized, on the continuum of periodic waves and white noise. For example, in the case of the sung text, vowel sounds are close to periodic waves and consonant sounds are close to white noise. He was able to make the boy's song as loud, as long, and as dense as his imagination required using the techniques of Musique Concrète. He arranged those sounds according to the degree of comprehensibility, and was able to make transformation from recorded sounds to electronic sounds and vice versa.

There are 11 elements of the electronic sounds, categorized as follows:

1. Sine waves
2. Periodically frequency modulated sine waves
3. Statistically frequency modulated sine waves
4. Periodically amplitude modulated sine waves
5. Statistically amplitude modulated sine waves
6. Periodic combinations of both types of sine wave modulation simultaneously
7. Statistical combinations of both types of sine wave modulation simultaneously
8. Colored noise with constant density
9. Colored noise with statistically changed density
10. Periodic sequences of filtered impulses
11. Statistical sequences of filtered impulses

This is also a scale between the continuum.

The colored noise is obviously noisy, but the statistical control also makes the sound more or less noisy. One more noise Stockhausen used in this work is the consonant sounds in the sung text. These are all used as elements in the scale.

It is in fact almost impossible to make spectrally continuous transformation between those two extremes or between two certain sounds with this technique. We could say, however, it is useful to make interesting and complex sounds within the context of serial composition.

Kontakte

In *Kontakte* (1959-60), the scale of timbres was made as in *Gesang der Jünglinge*. That is, the continuum between periodic waves and noise was considered also in this work. In addition, each element of the scale has its own scale of pitch. The pitch scale of the most periodic sound is divided into 42 tones per octave, and the noisier the timbre, the bigger the interval between pitches, because one cannot easily recognize a change of pitch of a noisy timbre if the interval is too small. He tried to make perceptual balance between the scale of periodic sounds and that of noise. Serial manipulation was applied to these scales.

When we talk about this piece, we should talk about the concept of 'unified time structuring' (Stockhausen 1991, pp. 91-96). Stockhausen described his experiment with pulse series as follows: "I recorded individual pulses... and spliced them together in a particular rhythm. Then I made a tape loop of this rhythm... and then I speed it up.... After a while the rhythm becomes continuous, and when I speed it up still more, you begin to hear a low tone rising in pitch.... You don't actually hear the rhythm any more, only a specific timbre, ... which is determined by the composition of its components."

Suppose we play the original rhythm with this timbre. Then the structure of the material and the form is exactly same. Only the speed is different. The traditional concept that implies timbre (instrument), melody, rhythm, and form are all different, is now changed. They are just different aspects of time perception.

With this experiment, Stockhausen also performed another experiment concerning noise.

According to his experiment, noise is defined as sound that has unstable periodicity. Stockhausen wrote: "If I change the periodicity of the sound: a little faster, a little slower, ... then the sound starts oscillating around a certain middle frequency, and all the half vowel or half consonant components, which are already fairly broad-band, begin to break up. So the continuum between sound of fixed pitch and noise is nothing more than that between a more and a less stable periodicity." Here he talked about random frequency modulation, which, in fact, cannot produce white noise. However, injecting randomness and controlling its degree is an important method to move between periodic waves and a certain degree of noise.

Conclusion

In this chapter, I briefly described Stockhausen's techniques as applied in two specific works. He did not rigorously apply these techniques as a mathematical formula. However, he has always depended on his ears to discover musical material which could interest him. His techniques thus have always been under the control of his musical sense.

4 Koenig's Noise

> Sounds with variable time-structure. Their oscillation curve moves between periodic and aleatoric sequences, and between sinusoidal and impulse-like wave-forms. ... The movement of the timbre rather follows the compositorial processes of the microtime range. In this way structures are composed instead of timbre.
>
> Koenig (1971)

According to Koenig, each conventional instrument has a fixed timbre, and this is an element of the language of instrumental composition. Composers should compose their music according to the syntax of this language. The various timbral modifications and extensions for conventional instruments, as seen in Chapter 1.2 and also mentioned in Chapter 3.1, reduced the limitations of such fixed timbre to a certain degree, but did not give the composer complete freedom.

In electronic music, to inherit this conventional language may bring more limitation in many cases. For example, the limitation of the number of available oscillators in the case of additive synthesis, and the absence of the irregular waveforms that appear in the startup vibration of a conventional instrument, brings simple timbre and the limitation of musical expression. Koenig said that electronic sound synthesis is thus only useful when a new language is created that does not inherit from the conventional language, such as the continuous transition or movement of timbre.

In Koenig's electronic music, timbre is not a musical parameter such as pitch, duration, and intensity, and is not a fixed property of a sound. Timbre is considered as the result of the relationship between the amplitude of a sound and time, and also the context (Koenig 1971). Composers design the inner structure of sound, especially change of the amplitude over time, but not the timbre. By doing this, the composer can freely explore the whole timbral space of sound, from sine waves to noise. Therefore, to Koenig, noise is not a characteristic of sound that a composer can choose, but a part of the *dimension* of timbre.

It is stimulating to examine this approach and hear complex noises as the result of Koenig's composition at the microlevel. In Chapter 4.1, *Funktionen* are taken as examples that show Koenig's serial technique applied to the structure of sound synthesis. Chapter 4.2 describes his computer program *SSP* which contains more systematic synthesis techniques. Both techniques make various degrees of noisiness in the resultant sound.

4.1 *Funktionen*

Funktionen is a series of compositions written by Koenig in which only two voltage curves are used as sound sources and at the same time as control signals with various combinations of electric devices. There are several versions such as *Funktion Grün*, *Funktion Gelb*, and so on, all of which are different in the musical forms but use the same materials.

As mentioned in the previous chapter, the timbre in these works is not treated as a musical parameter, but as the result of the composition of microstructure. It is interesting that most noisy timbres in these works are the result of the noise generator that is used as the control signal of a pulse series, not the noise generator used as a sound source.

Voltage Curves

In the *Funktionen*, there are two voltage curves which are made with "variable function generator" (VFG). The first is the "serial curve" which consists of 50 samples in a period (Figure 4.1.1 (a)). The second is "01-curve" which is the sequence of square impulses made with the alternation of the maximum amplitude (1) and the amplitude zero (Figure 4.1.1 (b)). These two curves are used as sound sources or as control signals for other devices. Koenig created five basic source materials which are described in the next section.

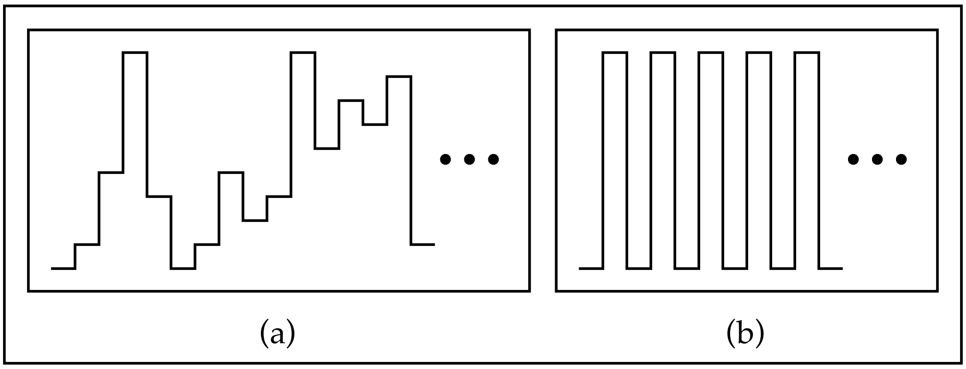

Figure 4.1.1 Two voltage control sources used in *Funktionen*.
(a) "serial curve" which consists of 50 samples.
(b) "01-curve".

Basic Materials

BASIS The serial curve (Figure 4.1.1 (a)) is used at high speed so that it makes an audible sound. The composer can control the speed to obtain different pitches, but only 13 frequencies are given.

MEL The serial curve modulates the frequency of a sine wave. The modulation frequency lies between 32Hz and 724Hz, which causes a complex sound rather than clear frequency changes.

PULS The serial curve makes audible sounds as in BASIS. Each sample, however, has an irregular duration because it is triggered by the peaks of a random signal. This material plays an important role in making noisy sounds in this work.

The above are used as the sound material for the *Funktionen* compositions. The sound material is combined and treated with the following transformations, ring modulation, filtering and reverberation to create two additional and composite sources of sound material.

RING The serial curve (partly filtered by means of the VFG's own filters for making the curve more triangular) modulates the frequency of a ring modulator. The modulation frequency lies between 100Hz and 1000Hz. The VFG is triggered by noise generator as in PULS.

MOD The serial curve (filtered as in RING but totally) controls an amplitude modulator. And this amplitude modulator controls the intensity proportion of a filter or reverberation device. The 01-curve is also used to turn it on and off alternately so that the amplitude modulated sound material is chopped.

Sound Production

The five materials described above are then combined into various sound production models. Figure 4.1.2 and Figure 4.1.3 show the examples.

Koenig uses 36 models in these works. For example, in *Funktion Grün*, each model is used four times. This work has 4 channels in which each model is used once. That is, the 36 models are differently organized in each channel.

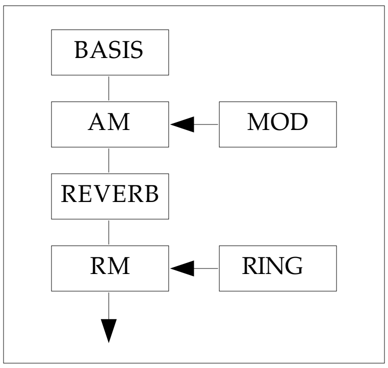

Figure 4.1.2 A sound production model (model 31). The basic material is amplitude modulated, reverberated and ring-modulated.

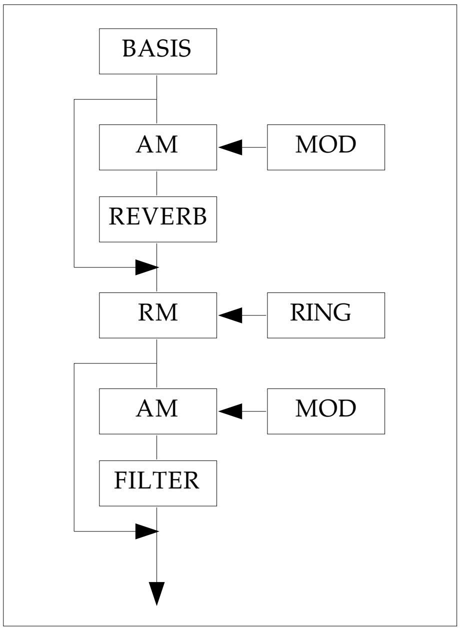

Figure 4.1.3 A sound production model (model 34). The basic material is amplitude modulated, reverberated, and mixed with the original. This is then ring-modulated, amplitude modulated, filtered and mixed with the previously ring-modulated sound.

Conclusion

The composer does not compose with fixed timbres, but rather composes the structures that produce the timbres. The serial manipulation of simple materials eventually makes various and complex timbres which are often noisy and rapidly change over time.
We can think about extensions. For example, the basic pulse series shown in Figure 4.1.1 is fixed, but we could vary it, possibly over time and obtain different kinds of noises. SSP which will be described in the next chapter is an extension designed by Koenig.

4.2 SSP

SSP (Sound Synthesis Program) is a computer program for sound synthesis designed by Gottfried Michael Koenig. The user or composer enters parameter values and the program produces sound in real-time. The parameters of the program are not explicitly musical ones such as pitch, duration, and intensity, but amplitude values and time values. Explicit musical elements are not set directly by the user, but rather are the result of the compositional rules at the microlevel.

Background

SSP has two main concepts behind it. The first is serialism. We examined in the previous chapter Koenig's techniques of serial manipulation over sound elements and material. That is, he composes sound using this compositional procedure instead of reconstructing a sound based on analytical data. SSP is a program that systematize this method. The other main concept is Koenig's computer programs *Project 1* and *Project 2*, which were created for instrumental composition. In these two programs the user specifies musical parameters such as pitch, duration and compositional or selection rules and the computer outputs a score with aleatoric processes. In SSP, the same rules are used to determine the microstructure of the sound with the controlled elements being amplitude values and time values.

Compositional Process

The process of composition with SSP is as follows: A series of time and amplitude values are specified (LIST). A selection is made from the list (SELECT). Segments containing time and amplitude values are formed (SEGMENT). These segments are arranged in a sequence (PERMUTATION). This arrangement of segments is performed (SOUND). The words in capital with parentheses are the *functions*.

SSP has a conversational user-interface. It asks the user questions and the user responds with entering a function and then entering the parameters of the chosen function.

Functions

LIST	An amplitude list and a time list are defined. These lists are source material for every other level of processing.
SELECT	A selection is made from the elements specified in LIST. A selection principle should be chosen.
SEGMENT	One or more segments are created. A segment is an equal number of amplitude values and time values chosen by SELECT.

PERMUTATION	The order of the segments is established.
SOUND	The segments are performed as ordered in PERMUTATION.
PRINT	The numerical results of various functions can be printed.
PLOT	The results of certain functions can be plotted.
STATUS	The number of elements in each file created by a function is listed.
HALT	The program stops.

Selection Principles

In the function SELECT, the user can choose one of the available selection principles. Most of these are based on aleatoric process which can make complex and noisy sounds.

ALEA	N random values are chosen between A and Z.
SERIES	N random values are chosen between A and Z. Each value between A and Z should be chosen once before all the values are chosen.
RATIO	N random values are chosen between A and Z according to the given probability distribution over the values between A and Z.
TENDENCY	N random values are chosen between boundaries that change in time. M number of masks may be specified. NN values are chosen in one mask that is indicated by initial boundaries (A1, A2) and final boundaries (Z1, Z2).
SEQUENCE	A sequence of numbers is given in order.
COPY	The available values are copied.
GROUP	A random value between A and Z is chosen, and this value repeated one or more times in succession to form a group. The size of the group is a random value between LA and LZ. The value between A and Z as well as the group size between LA and LZ may be chosen with either ALEA or SERIES. The combination of ALEA and SERIES is indicated with TYPE. In total N values are selected.

Sound of SSP

This program does not directly allow the composer ty input musical parameters, but its various selection principles offer a design tool of a different perspective. This, of course, requires a different approach to composition. There is a problem in relating input (chosen selection principles and parameters) and output (pitch, duration, intensity, timbre, etc.). One can approach to this problem by characterizing the input data (Berg et al. 1980). The output timbre in particular cannot be easily described in terms of th input data. However, we could characterize the sounds of SSP in general terms as follows:

Microtonality : One usually does not consider pitch when choosing the time value of a sample. Instead, it is chosen from the given elements of time values by a selection principle. Thus, SSP provides a continuous pitch scale which has nothing to do with the traditional one which divides an octave into 12 steps.

Emergence of Noisiness : The compositional rules in SSP are either various aleatoric procedures or the direct enumeration of a sequence of values. The composer, thus, can create sounds from a completely periodic wave to white noise. Usually there emerge various degrees of noisiness in the output timbres when combining those two types of compositional rules.

Conclusion

SSP is not well suited to a composer who must have a particular sound. Instead, SSP offers an interesting and experimental framework for a composer who wants to define structures and listen to the results.

5 My Noise

The synthesis methods described here have been developed by personal experimentation and experience during recent years. All of them are based on time domain process. The advantage of time domain synthesis is, as demonstrated by Xenakis and Koenig, the ease with which it's possible to explore the undiscovered areas of sound and the continuous area between musical tones and noise.

Experiments with mathematical formulae and with algorithms were key methods for me to create and develop these methods. These ways of experimentation were pleasant for me because of my interest in computer programming and in mathematics.

Chaos and Noise

The first experiments have been done using chaotic functions or "maps", which are mentioned in Preface of this paper. Logistic Synth is the computer program I have developed for this experiment. The program and a sound example are in Composing Noise CD ("Sound 01 Logistic Map" file or audio track 2). However, I will not describe these experiments in detail in this paper, because they have no musical control yet, and are still under experiment. However, these experiments made me become interested in noise.

Noise AM and FM

The second experiments have been done by introducing randomness into a sound synthesis system, such as frequency modulation synthesis (FM) and amplitude modulation synthesis (AM). In most cases, the injected randomness makes the system produce noise. For example, if we randomly modulate the amplitude or frequency or both in a sine wave generator, we can hear noise. It is not easy, however, to discover a method which produces stimulating noise which may be musically useful. In these experiments with AM and FM, control over the speed or degree of random modulation does not affect the sound greatly. We merely hear a clear sine wave with noise in background. We need to consider what may be added or removed. For example, using a low pass filter for the random modulator provides a much better result. The use of randomness in a sound synthesis system needs to be carefully considered if it is to be successful. There are sound examples in the CD; "Sound 02 Noise AM" and "Sound 03 Noise FM" files, or audio track 3 and 4.

After these early experiments, I began to think about rather complicated algorithms in which randomness is integrated. Xenakis' Dynamic Stochastic Synthesis became the most exciting synthesis model to me. One can find techniques such as interpolation and random walk in my algorithms as found in Xenakis' algorithm. I also used a few sound processing techniques such as digital filtering and vocoder (or convolution). They will be described in detail in the next chapters.

5.1 Noise Interpolation Synthesis

The algorithm described in this chapter uses random values and an interpolating function. Random values are directly used as the sample values of sound, but with a certain time interval between them. The samples between these values are calculated by a given interpolating function. While random values produce noise, the interpolated samples between them add a certain timbre into the noise.

Interpolation

Suppose we know the value of a function f(x) at points x1, x2, and x3, but do not know its value at point x4 because we do not have an analytic expression for f(x) that lets us calculate its value at an arbitrary x. In this case we can use all or parts of the values at the points x1, x2, and x3 to guess the value at the point x4. This technique is called *interpolation* if x4 is in between the largest and smallest of the points x1, x2, and x3, and it is called *extrapolation* if x4 is outside that range (Press et al. 1993).

Interpolation technique (from here on, extrapolation will not be considered because the synthesis techniques described in this chapter do not use it) have been applied to sound synthesis and signal processing because of the digital manipulation of sound. It has mostly been used for reducing or replacing complex calculations. For example, *table-lookup synthesis* calculates the sample values (e.g. sine wave) using a certain interpolating function with a table that contains a number of precalculated values (Dodge 1997).

Interpolation techniques, however, can be a useful tool for creating an abstract synthesis model because we can think of interpolation as the modeling of a function. *Noise interpolation synthesis* creates an abstract model that uses random values and interpolation between them.

The Algorithm

As explained above, random values are directly used as the sample values of sound, but with a certain time interval between them. The samples between these values are calculated by a given interpolating function. In this algorithm, two things are important: the random values and the interpolation function.

The use of various distribution functions for the random number generator will greatly influence the characteristics of the sound. In my implementation, however, only the uniform distribution is used, and instead, a second order feedback filter is used. This will be discussed later.

The use of various interpolating functions make different timbres of noise. For example, linear interpolation makes light and bright noise, half-cosine interpolation makes smooth sounding noise, and rectangular interpolation makes heavy and rough sounding noise. Figure

5.1.1 shows an example waveform in which half-cosine interpolation is used.

The change of the interval between random values or "frequency", also influences the timbre. Figure 5.1.2 shows the change of spectrum according to that of the frequency. The lower frequency, the less energy in high frequency components.

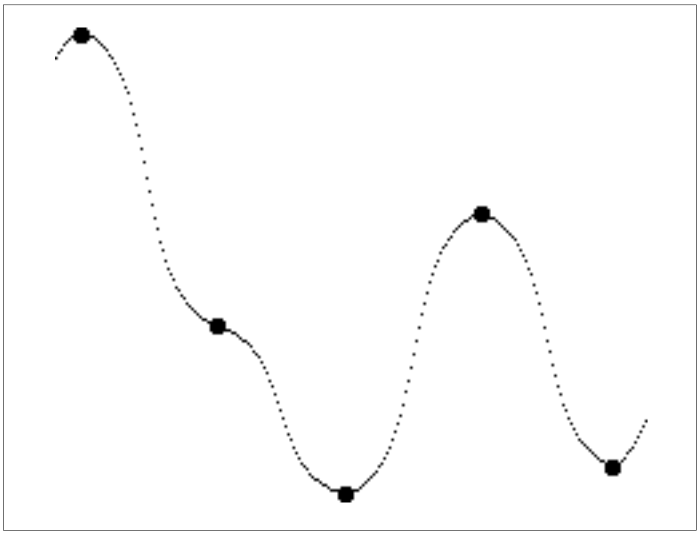

Figure 5.1.1 Noise interpolation synthesis : sample points represented by large dots are randomly generated and samples between them are interpolated by half-cosine function.

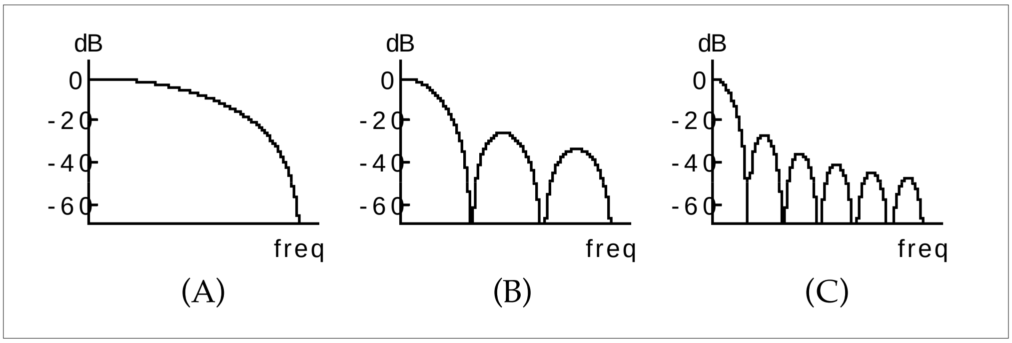

Figure 5.1.2 The spectra of noise interpolated sound (Figure 5.1.1). The higher the frequency of random points, the more high frequency components there are in the sound and the clearer pitch. Assuming the sampling rate is 12 kHz:
(A) Frequency = 6 kHz (1 interpolated sample between random values),
(B) Frequency = 2 kHz (5 interpolated samples),
(C) Frequency = 1 kHz sampling rate (11 interpolated samples).

Implementation : Trebari

Trebari is a computer program which implements noise interpolation synthesis. Trebari runs on a Power Macintosh computer and generates noise in real time. The user can control the parameters of Trebari with MIDI signals. You can find Trebari in the Composing Noise CD.

Trebari consists of *filtered random number generator*, *dynamic interpolator* and *processor* as shown in Figure 5.1.3.

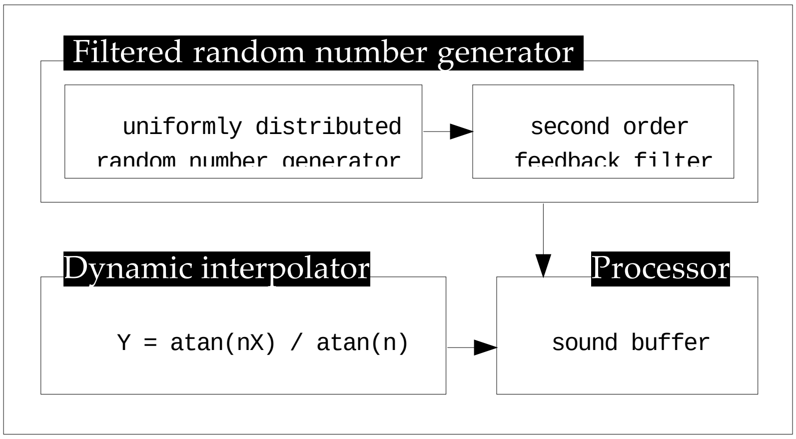

Figure 5.1.3 The block diagram of Trebari.

<u>Filtered random number generator</u>

This consists of two parts; a uniformly distributed (pseudo-)random number generator and a second order feedback filter. The filter is used to give some periodicity to the random numbers. This does not mean that a certain amount of random numbers will be repeated, but this means a certain complex correlation will be introduced into the random numbers. I used this to change the characteristic of randomness, while Xenakis used various distribution functions (Chapter 2). The user can control the degree of filtering. Note that this is not same as subtractive synthesis which filters a sound, or noise, since Trebari does not filter its output sound, instead it filters the random numbers which is the primary synthesis data. This can bring more complex periodicity in the output sound in combination with the frequency of random values. This will be discussed later in this chapter.

<u>Dynamic Interpolator</u>

The interpolating function in Trebari is not static but dynamic, which means the user can change continuously from one interpolation function to another while sound is being synthesized. For this feature, an arc-tangent function is used. Eq. 5.1.1 is the actual function used in the program. The numerator of the function is the main contribution to the interpolation. The denominator is a normalizing element that confines the output range on the x-axis within -1 and +1. According to ‘n’ in the equation, the type of interpolation is changed continuously from linear to rectangular (Figure 5.1.4).

Processor

This puts the generated random values and interpolated values into a sound playback buffer. The engine for real time sound processing on a Macintosh computer is RTSS (Real Time Sound Server) which was written by myself in the C++ programming language. This package is included in the *Composing Noise CD*.

$$Y = \frac{\mathrm{atan}(nX)}{\mathrm{atan}(n)} \qquad (n > 0,\ -1 \le X \le 1\ ;\ -1 \le Y \le 1)$$

Eq. 5.1.1 Dynamic interpolation function as used in Trebari.

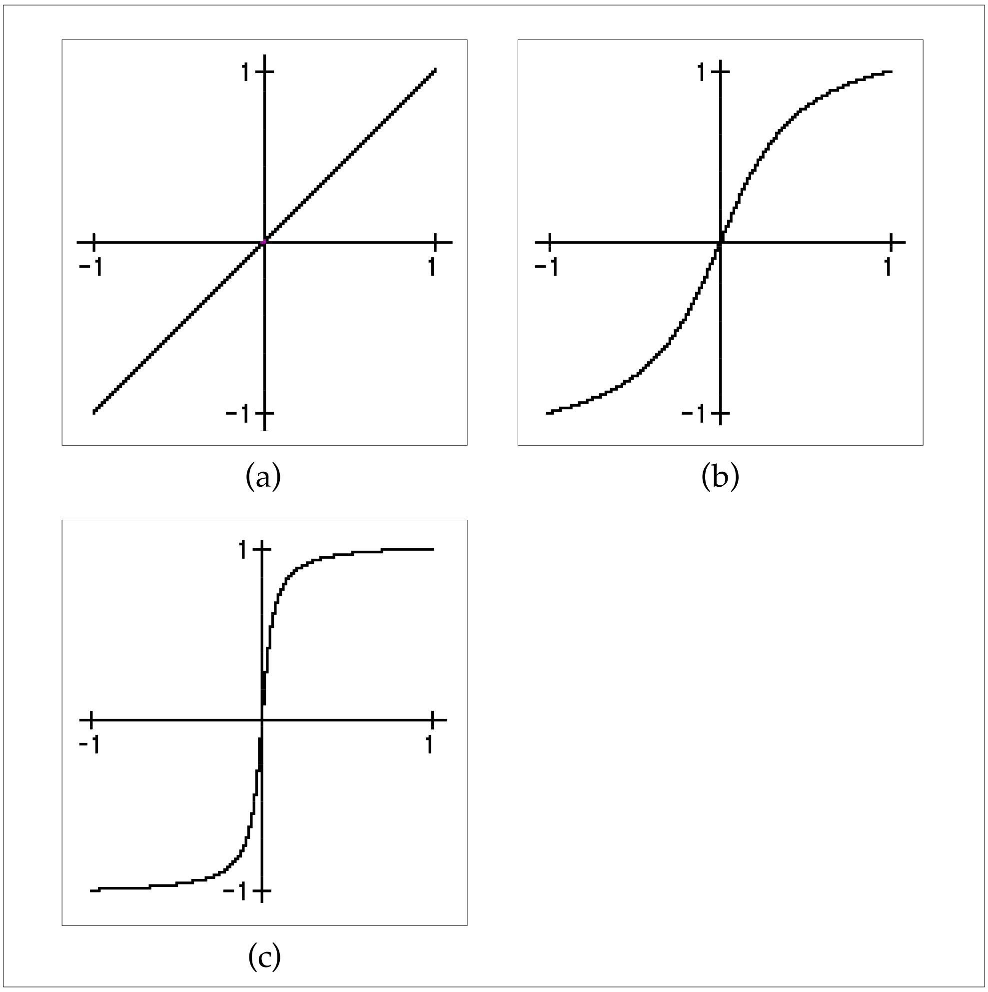

Figure 5.1.4 The graph of the dynamic interpolation function (Eq. 5.1.1).
(A) n = 0.01 , (B) n = 3 , (C) n = 20

Parameters and Noise of Trebari

The parameters of Trebari and their influence over the sound are described here.

<u>Frequency</u>

This determines the time interval between two random samples. At low frequency, the noise of Trebari has less high frequency content and higher frequency there is significant high frequency content. As this parameter is changed, you can hear quite clearly the change of pitch, especially at low frequencies, because the sound has a group of noise bands with an harmonic relationship (Figure 5.1.2).

<u>Frequency Random Deviation</u>

This randomly distorts the Frequency parameter, which means it distorts the harmonic relationship among the noise bands mentioned above. This parameter is useful when the composer wants a more noisy sound. This value is specified in Hertz. If the frequency is 200Hz and this value is 50Hz, the actual frequency randomly varies between 150Hz and 250Hz. If you set a larger value for 'frequency random deviation' than for 'frequency', for example 300Hz and 200Hz respectively, the sound will have sudden holes in it, because the actual frequency can go below the minimum audible frequency.

<u>Amplitude</u>

The amplitude of the sound.

<u>Pan</u>

The panning value. The current version of Trebari provides only one oscillator but the sound is sent to two channels. As you change this parameter value, the panning between the channels is changed.

<u>Roughness</u>

This determines the shape of the interpolating function. This directly sets the 'n' in Eq.5.1.1. We have already seen the graph of this equation in Figure 5.1.4. The larger this value, the rougher the sound we obtain.

<u>Resonating Frequency for RNG</u> & <u>Resonating Magnitude for RNG</u>

The frequency response of a second order feedback filter has one peak with a certain bandwidth (Figure 5.1.5). The Resonating Frequency (RF) parameter changes the center frequency of the peak and the Resonating Magnitude (RM) parameter changes the magnitude of the filtering. The RF parameter takes effect only when the RM parameter has enough magnitude to provide a resonance peak. I have already mentioned that this filter is not applied to the output sound, but to random numbers. The influence on the output sound is, however, similar to filtering the noise. When the RF parameter is a low value, the low frequency components of the sound are emphasized. Conversely, when the RF has a high value the high frequency components are emphasized. This gives an additional effect which is similar to ring modulation in combination with the Frequency parameter. It is possible to hear such effects, especially when the RF parameter is changed continuously. Thus, the timbre of the sound depends on both of the parameters Frequency and the RF. For example, with a low frequency and a high resonating frequency it sounds like a filtered noise whose energy is very unstable. With a high value for frequency and the same resonating frequency it sounds like a rough voice.

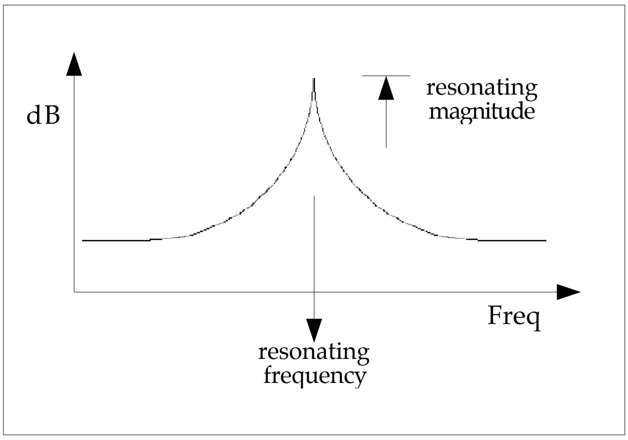

Figure 5.1.5 The frequency response of a second order feedback filter and the parameters.

Conclusion

The sound of Trebari suits dramatic music. I would say that two edges of its sound space are relaxed noise and tense noise. Its parameters can make the sound move between these two points. The *Improvisation for Noise Solo* described in the next chapter is based on this sound space. The Composing Noise CD contains a series of some fragments of the piece, which is an sound and music example of Trebari ("Sound 04 Trebari" file or audio track 5).

However, this sound space and thus also the synthesis method is too limited for general composition. It can server a purpose to play a specific role in a layer of a music, it is a specific and somewhat unique voice.

5.2 *Improvisation for Noise Solo*

Improvisation for Noise Solo is my work for the monophonic sound of Trebari which is described in the previous chapter. This chapter explains how this wok has been constructed. This work was performed at 3rd Sonology Discussion Concert in March 1998.

Equipment Overview

In this work, two computer programs (Trebari and Max) and two MIDI devices (a MIDI fader box and a MIDI keyboard) are used. The block diagram of the setup of this equipment is Figure 5.2.1.

MAX is a real time MIDI processing program from IRCAM and Opcode Inc. This program assists the performer of this work by manipulating and assisting the control and flow of MIDI data. Since it is difficult and restrictive of musical expressions for a solo performer to control all of the parameters of Trebari at the same time, Max controls some parameters.The performer controls both the parameters of Trebari and Max with the MIDI fader box and the MIDI keyboard. Max also controls some parameters of Trebari. The flow of these MIDI signals are shown in Figure 5.2.1.

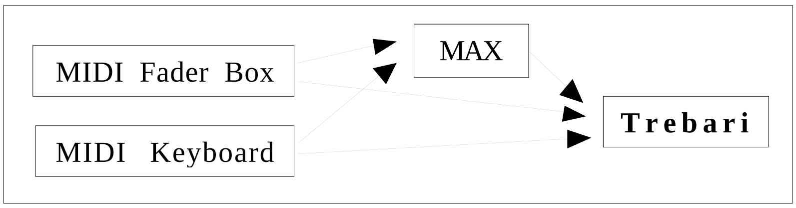

Figure. 5.2.1 The block diagram of the equipments used in this live performance and the flow of MIDI signals.

Construction

This work has been constructed via various experiments with constant review and feedback from the achieved results. First the plan for a section was briefly sketched and described on a paper (for example, section 1 : play loud and high frequency noise for 1 minute with as small changes as possible). Then that planned section was improvised. Reviewing the results of the improvisation would lead to a modification of the plan to achieve a more musically satisfying section. Initially, I did not use Max in the process but only the MIDI fader box and MIDI keyboard to control each parameter of Trebari.

This performance method, however, required long practice to achieve the required musical expression and those expressions not of the required standard. Thus, Max was added to control some parameters in a programmable, but still controllable manner. For example, switch 1 on the MIDI fader box triggers Max to make the amplitude parameter of Trebari increase and decrease in a programmable and controllable amount of time.

After exploring various Max patches, the structure was divided into 6 sections, each with specific and planned characteristics. After considerable experimentation, the completed plan is as follows:

Section 1

- Explosive expression (with slow change of Amplitude and Roughness).
- Faster and faster changes of Amplitude and Roughness.
- Static, a little amount of filtering with low Resonating Frequency (to have more low frequency components).
- (No MAX patch used)

Section 2

- Low frequency with Gaussian envelope of Amplitude.
- Random Argument and Magnitude according to movement of Amplitude (MAX).
- Overall volume is soft with some silence parts.

Section 3

- High frequency with many changes.
- Gaussian envelope of Amplitude on every note-on (MAX) + manual changes of Amplitude and Roughness (Performer) = makes conflict between two.

Section 4

- Random-walk of Frequency (MAX).
- Performer controls the range and the step size of the random-walk.
- Performer also controls Amplitude, Roughness, and filtering.

Section 5

- Very loud and rough noise with a little bit change of Frequency (high) and filtering.
- (No MAX patch used)

This plan (or "score") has been rehearsed and performed at the previously mentioned concert. There is no duration specified for each section, but through the rehearsals durations were chosen for each section that were musically appropriate. In the first concert, however, the performance lasted a little less time than I had expected.

Conclusion

This work is my first live performance. I have become much more interested in the live performance of electronic music while creating this work. I have also discovered that it is quite interesting and exciting to make electroacoustic music through the intuitive feeling processes of a composer. What needs considerably more development and research is the development of higher level control parameters for various musical expressions. Also, an investigation is required into appropriate controllers that provide a correlation between the gesture of the performer and the resultant sound.

The parameters of Trebari are too primitive to make varied and complex musical expressions in the context of a live performance. This difficulty is conquered by making a higher level of control using Max. For example, in the section 4 of this work, MAX generates a random walk of the frequency parameter and the performer controls its step size and range.

In this work, there is a conflict between the physical gesture of the performer and the audible result. While the sound was sometimes explosive, the performer's gesture was very small. This is partly because of the small sliders of the MIDI fader box, but perhaps a different controller would be better. It would add an extra visual effect if a controller was used that allowed for better mapping from the physical gesture to the audible result. This requires considerable further research and development.

5.3 Miscellaneous Techniques

The techniques described in this chapter list some of my further experiments with the use of randomness in sound synthesis.

Two Extended Versions of Trebari

The limited sound space of the Trebari program led to investigations of extensions to the technique. The vocoder (described below) was considered to be potentially highly useful in extending the timbral range of possibilities because it allows for another signal input. This led to the development of two different vocoder versions of Trebari, one uses a sine wave for the second vocoder input and the other allows live sound input. The former is named *Trebari Vos* (VOcoder with Sine wave) and the latter *Trebari Live* (Figure 5.3.1).

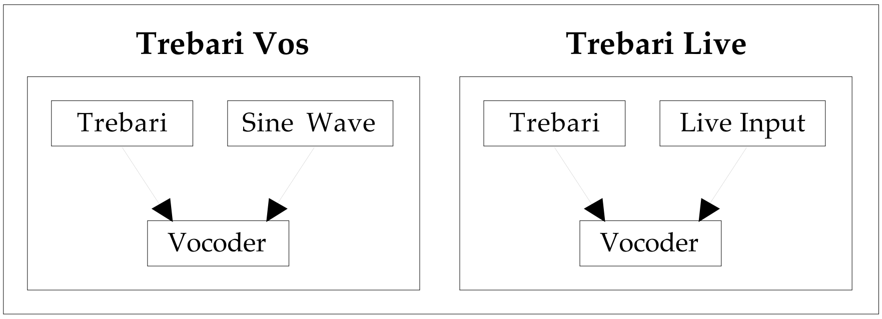

Figure 5.3.1 Two Extended Versions of Trebari using Vocoder

<u>Vocoder</u>

The vocoder is also called a subtractive analysis/synthesis system, which was originally built for the data reduction of speech signals (Roads 1996, pp. 197-199). Simply speaking, it puts the spectral envelope of sound A on the spectrum of sound B. This results a spectral fusion.

The original vocoder model consists of an analysis part that extracts the spectral envelope from sound A and a synthesis part that applies this spectral envelope to sound B. Each of these two parts contains a group of band pass filters. This process requires considerable calculation and thus it is calculation intensive for a real-time application such as Trebari.

To achieve the required performance for real-time signal synthesis and processing, a faster algorithm was implemented. A vocoder can be considered as a convolution between sound A and sound B (one of them can be considered as an impulse response). The convolution algorithm (Roads 1996, pp. 419-426) is implemented instead of a vocoder algorithm. In this implementation the whole sample set of sound A or B is not used for one calculation, but rather a small (windowed) subset of samples are used for faster calculation.

<u>Trebari Vos</u>

I consider a vocoder as a tool which can easily make an abstract transition between two sounds. By 'abstract transition', I mean that the path of the transition is an 'unknown' space between two sounds. Thus, the perceptually perfect transition between two known sounds, for example, flute and piano, is not concerned here.

I took a sine wave as the second input for the vocoder. My intention was to make an abstract continuum between a sine wave and Trebari noise (Composing Noise CD has an example; "Sound 05 Trebari Vos Continuum" file or audio track 6).

The use of a sine wave adds two more parameters (Figure 5.3.2), the frequency and amplitude of the sine wave. The sine wave now plays a major role in the sound and Trebari plays the role of introducing noisiness into the sine wave. The Frequency parameter of the sine wave generator determines a pitch and the Frequency parameter of Trebari determines the degree of noisiness.

The timbre range of Trebari Vos extends the original and ranges from a sine wave to the typical noise of Trebari. The smaller the Frequency parameter value of Trebari, the closer is the sound to a sine wave and vice versa. Trebari Vos has greater timbral range than Trebari and is more useful in composition.

I wrote *Etude* No.1 and No.2 for Trebari Vos. I used MAX (the same program as used in *Improvisation for Noise Solo*) to control all the parameters of Trebari Vos according to score files each of which contains functions and values for a parameter. “Sound 06 Etude No.2” file or audio track 7 is *Etude* No.2 for three Trebari Vos oscillators.

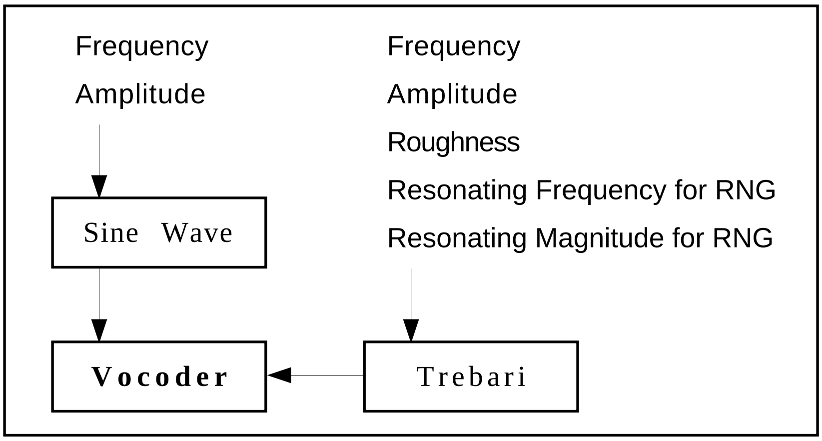

Figure 5.3.2 The parameters of Trebari Vos.

Trebari Live

This version was developed by replacing the sine wave of Trebari Vos with live input. My intention was to obtain different sound continua. Since any sound can be used for the live input, one can obtain various continua between a sound and noise.

Trebari Live also runs on a Power Macintosh computer. The user can use the built-in input of a Macintosh computer, an audio file, or the input of an installed audio card. The input sound is convolved with the Trebari signal, which results in timbral change.

As in Trebari Vos, the frequency parameter of Trebari plays the important role of introducing noisiness. The smaller this value, the clearer the original sound. As expected, the larger greater the Trebari frequency value, the noisier is the output sound. In Trebari Live, the Resonating Frequency and Resonating Magnitude parameters (of the feedback filter) are especially important in changing the timbre of the original input sound. They provide a broad range of effects and timbral modification.

Trebari Live is not a synthesis tool, but rather a sound processing tool. This kind of system can supply the musical gesture of the original input sound and also change the timbre completely. This is useful for composition and also in context of live performance.

Conventional instruments might be good for the live input, because various and intuitive expressions of the players can be easily obtained while the instrumental timbres are transformed into complex noises (“Sound 07 Clarinet and Cello” file or audio track 8 is an example).

Noise Filtering : Fried Noise

Digital filter theory is another subject in which I have been interested. A few years ago I was curious what would happen to a source sound if a filter controlled by a random process was applied to the sound. Recently I have realized this.

Noise Filtering is based on subtractive synthesis which filters noise to change its spectrum. In this algorithm, a band pass filter is used, whose resonating frequency and magnitude randomly move within given ranges (Figure 5.3.3). The noise used in this algorithm is an interpolated noise as in Trebari. However, only half-cosine interpolating is used.

In this algorithm, there are two components which produce noise. One is the source sound (half-cosine interpolated noise), and the other is random walks of the filter parameters. The speed and the step size of the random walks determine the degree of noisiness produced by the filter.

The sound of this synthesis algorithm can be conveniently described in everyday terms. With a small step size and range, it sounds as if a noise is 'boiling', when the step size and range is large it sounds as if the noise is being 'fried'. Thus, this algorithm is named Fried Noise. The sound of this algorithm can be used as an effect, but is not generally useful in a musical context. (Sound example : "Sound 08 Fried Noise" file or audio track 9.)

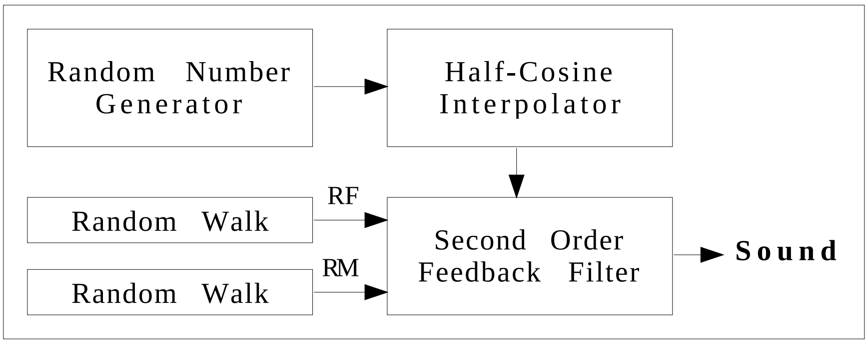

Figure 5.3.3 The block diagram of Fried Noise. 'RF' stands for Resonating Frequency, and 'RM' for Resonating Magnitude.

Dynamic Random Waveshaping : Rasin

The most recent development of mine is *Rasin*, which combines a wave shaping technique with random process.

Wave shaping is a synthesis technique that creates a complex sound with simple calculations (Roads 1996, pp. 252-260). Starting with a sine wave, it is possible to create many different waves by using different "shaping" functions.

In this implementation a dynamically moving shaping function has been implemented instead of a fixed one. In this process there are segment endpoints in the boundary of the shaping function area which randomly walk and the continuous wave shaping function is made with linear interpolation between them (Figure 5.3.4). This idea is adopted from the algorithm of Xenakis' Dynamic Stochastic Synthesis (Chapter 2.1).

This algorithm has three parameters; the *frequency* and *amplitude* of the sine wave generator and *step size* of the random walks for the segment endpoints in the wave shaping function. The resulting sound is a periodic wave whose spectrum is constantly changing over time. The larger the step size, the clearer and faster are the changes of the spectrum and the noisier high frequency components. This algorithm is called Rasin synthesis from RAndom SINe wave synthesis. (Sound example : "Sound 09 Rasin" or audio track 10, and a music example : "Sound 10 Etude No.5" or audio track 11.)

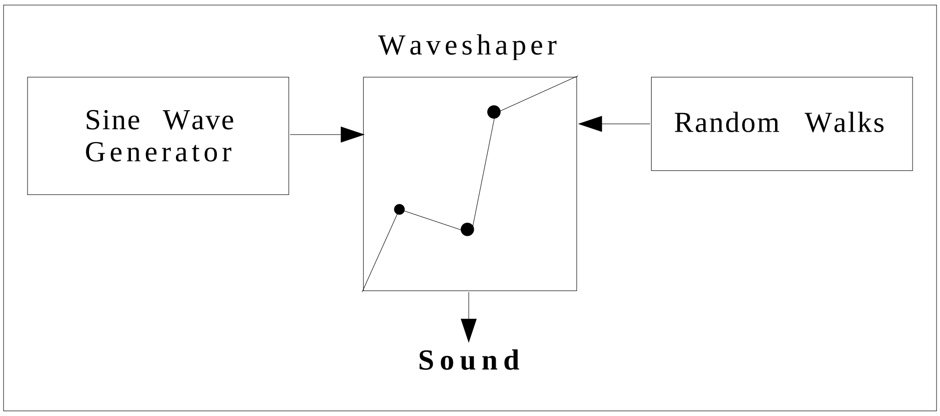

Figure 5.3.4 The block diagram of Rasin. In the wave shaping function, there are endpoints (3 in this example) and they are linearly interpolated.

Conclusion

The techniques described in this chapter are not general tools for general composition or synthesis. They are, rather, more like instruments which produce a certain range sound timbres. I have tended to focus on algorithms and implementations for sound synthesis. However, now my interest leads me to consider the more important question of how to compose with these elements.

Taking the philosophical viewpoint that composing noise or sound is the same activity as composing music, then these algorithms should be extended such that the various parameters may evolve over time. This activity will be undertaken in the near future.

Conclusion

In this paper, we have seen how Xenakis, Stockhausen, and Koenig composed noise. By 'composing noise' I do not only mean synthesizing various noises, but I also mean the integration of sound synthesis and macrolevel structuring which can explore the continuum of periodic waves and noise.

This is obvious in the case of Xenakis. In his Dynamic Stochastic Synthesis model, a stochastic process makes a waveform evolve over time. The stochastic process here is thus a part of an algorithm and at the same time, a macrolevel structuring process.

In the case of Stockhausen, there was no fixed model for sound synthesis. In fact, it seems that he did not like to develop low level synthesis method. He usually used sine wave generators, noise generators, and their combinations, and recorded sounds. The serial techniques were applied both to the use of such electronic generators and to the macrolevel structure.

In Koenig's SSP, various selection principles and their combinations arrange primitive materials (amplitude and time values) to build up a microlevel structure (waveform), and arrange waveforms to build up a macrolevel structure.

I think these are very important and interesting aspects of their attitude about noise and structuring. In my algorithms shown in Chapter 5, this is currently missing. As I wrote in the conclusion of Chapter 5.3, the concept of 'unified structure' should be interesting to experiment.

Appendix

A. About Composing Noise CD

This paper is accompanied by “Composing Noise CD”. There are two formats in this single CD; Audio CD (which can be played in any standard CD player) and HFS (the standard Macintosh file system, which will be represented as a “volume” when launched on a Macintosh computer). What each part contains is as follows:

Audio CD part (general audio tracks)

- Sound examples : The audio tracks from 2 to 11 on this CD contain sound examples which are referred to from some parts of this paper.

- Compositions : The audio tracks from 12 to 15 on this CD contain my compositions:
  - *Improvisation for Noise Solo*
  - *Etude* No.5
  - *Let Them Talk* (part)
  - *Improvisation for Noise Instruments* (from a rehearsal)

* **Important Note** : There is no audio on the track 1. It contains data of HFS part. If you play this CD from a standard CD player, you will hear nothing from the track 1. This is why the sound examples start with track number 2.

If you play this CD on a Macintosh computer using AppleCD Audio Player, you will not see the data track in the program. Thus the audio track 2 becomes track 1 in the program. You have to consider this not to be confused with track numbers. So I would recommend you to play sound files instead of audio tracks, if you use a Macintosh computer.

HFS part

- A PDF (Portable Document Format) file of this paper.

- Sound examples : These are the same as those in the audio tracks, but in AIFF (Audio Interchange File Format). One can choose one of them at his/her convenience. To open an audio file, an AIFF player software such as SoundApp is needed.

- Compositions : The AIFF files, the documentation, and the related files such as MAX patches of my compositions are available. The compositions included here are :
    - *Improvisation for Noise Solo* (a live performance for Trebari)
    - *Etude* No.5
    - *Let Them Talk*
    - *Improvisation for Noise Instruments*

- Software : These are the Macintosh programs I have developed for the projects described in this paper.

B. Sound Examples

Here are the descriptions of the sound examples on the Composing Noise CD.

<u>Audio Track 2</u> (file "Sound 01 Logistic Map")
This sound was made with a sine wave whose frequency was modulated by a chaotic or iterative function called "logistic map". The function is as follows:

$$N = r \cdot Np \cdot (1 - Np)$$

where N is a new output value and Np is the previous value (the initial N value should not be zero, and N and Np range from 0 to 1), and r is a given constant (which ranges from 0 to 4). All the values are real numbers.

The constant r largely influences the characteristic of an output series of values. Only when r is larger than 3.57, it shows a chaotic behavior which seems random, otherwise, the output series of values settles down to a single value or becomes periodic.

Each output value was mapped into a value in hertz (the range is given), which was used as the frequency value of a sine wave.

This sound example consists of two sounds. For the first one, the initial N was 0.3 and r was 3.754, and the range of mapping was from 354 Hz to 455 Hz, and the modulating frequency was 111 Hz. For the second one, the initial N was 0.42 and r was 3.912, and the range of mapping was from 1100 Hz to 2433 Hz, and the modulating frequency was 975 Hz.

<u>Audio Track 3</u> (file "Sound 02 Noise AM")
This sound was made with a sine wave whose amplitude was modulated by random values. In this sound, the frequency of the sine wave and the range of the modulation are fixed, but the modulating frequency increased with time. Thus you will hear amplitude changes clearly at the beginning, but those changes will make the sound noisier with time.

<u>Audio Track 4</u> (file "Sound 03 Noise FM")
This sound was made with a sine wave whose frequency was modulated by random values. In this sound, the modulating frequency increased with time. Thus you will hear frequency changes clearly at the beginning, but those changes will make the sound noisier with time.

<u>Audio Track 5</u> (file "Sound 04 Trebari")
This was adopted from my live performance "Improvisation for Noise Solo", and edited to show the sounds and musical possibilities of Trebari (please refer to Chapter 5.1 and 5.2).

<u>Audio Track 6</u> (file "Sound 05 Trebari Vos Continuum")
This is a sound example made with Trebari Vos. This shows an abstract continuum between a sine wave and noise, which Trebari Vos algorithm can produce (Chapter 5.3).

<u>Audio Track 7</u> (file "Sound 06 Etude No.2")
Etude No.2 is a composition which was made with Trebari Vos. I made score files for MAX (a real time MIDI processing program for Macintosh computers) each of which contains functions and values for a parameter. Thus all the parameters were controlled by MAX according to the score files. The MAX patches and the score files are included in "Etude No.2" folder in "Compositions" folder on Composing Noise CD.

<u>Audio Track 8</u> (file "Sound 07 Clarinet and Cello")
This is a sound example of Trebari Live (Chapter 5.3). For the live input, a clarinet and a cello were used. This was a record of improvisation of the instrument players and a Trebari Live player.

<u>Audio Track 9</u> (file "Sound 08 Fried Noise")
This is a sound example of Fried Noise (Chapter 5.3). In this example, the frequency and the step size increases with time.

<u>Audio Track 10</u> (file "Sound 09 Rasin")
This is a sound example of Rasin (Chapter 5.3). In this example, all the parameters are not changed for the first 11 seconds to show the effect of dynamic wave shaping, and then the step size parameter is randomly changed for the next 9 seconds.

<u>Audio Track 11</u> (file "Sound 10 Etude No.5 (part)")
This is a musical example of Rasin. In *Etude* No.5, Trebari Vos and Fried Noise were also used. In this example (a part of *Etude* No.5), however, Rasin plays a major role. There are 16 voices of Rasin in this example.

C. Compositions

Some of my work is on the Composing Noise CD. Here is a brief introduction:

Improvisation for Noise solo (1998) [audio track 12]

This is a live performance piece for the monophonic sound of Trebari. Music was improvised by the composer according to a predescribed plan.

Etude No.5 (1999) [audio track 13]

This is an electroacoustic music, which consists of a number of voices of three noise generators (Trebari Vos, Fried Noise, and Rasin). Parameters scores were constructed, and realized with Jaeho Complex program.

Let them talk (1999) [audio track 14]

This is an electroacoustic music, which is not finished yet. The structure, the sound, and the way of construction are based on Etude No.5. This CD contains the first 7 minutes of music.

Improvisation for Noise Instruments (1999) [audio track 15]

This is a live performance piece for Clarinet, Violoncello, and Trebari Live, which is not performed yet. The record of a rehearsal is included. Please understand the quality of both sound and music is not good.

* *Let them talk* and *Improvisation for Noise Instruments* are performed on the final examination concert.

D. Software

The CD includes software which I have developed for the projects described in this paper. All the programs run only on MacOS™. Here is a brief introduction:

Trebari

Trebari is a computer program which implements noise interpolation synthesis (Chapter 5.1).

Trebari Vos

Trebari Vos is an extended version of Trebari. It uses a vocoder to spectrally fuse two sounds, a sine wave and Trebari noise. This is explained in detail in Chapter 5.3.

Trebari Live

Trebari Live is the same as Trebari Vos, but allows live input instead of a sine wave. This is also explained in detail in Chapter 5.3.

Xenak

Xenak is an implementation of Dynamic Stochastic Synthesis developed by Iannis Xenakis (Chapter 2.1). This implementation has only "barrier" control.

Jaeho Complex

Jaeho Complex is a non-real-time synthesis tool. The user writes a score file, and Jaeho Complex compiles it and makes an audio file. This program has been developed for experimenting with various synthesis methods, including Trebari Vos, Fried Noise, and Rasin. Thus, this is not a general tool, but experimental one for myself.

Real Time Sound Server

Real Time Sound Server (RTSS) manages all the basic tasks for a programmer to make real time sound processing on Apple Sound Manager. RTSS consists of a main class (SoundServer) and some assistant classes. They are all written in the C++ programming language. RTSS was written for my projects, and later released on the Internet. All of my real-time programs such as Trebari, Xenak, and so on, use RTSS as a real-time sound engine.

References

Bartolozzi, B. 1971. *New Sounds for Woodwind*. London: Oxford University Press.

Berg, P., Banks, J.D., Rowe, R. Theriault, D. 1979. *SSP - A Bi-parametric Approach To Sound Synthesis*. Utrecht : Institute of Sonology.

Berg, P., Rowe, R. Theriault, D. 1980. "SSP and Sound Description" *Computer Music Journal* 4(1). Cambridge, Massachusetts: The MIT Press.

Cage, J. 1967. *Silence*. Second Edition. Cambridge, Massachusetts: The MIT Press.

Cope, D.H. 1993. *New Direction in Music*. Sixth Edition. Illinois: Waveland Press Inc.

Cowell, H. 1930. *New Musical Resources*. New York: Alfred A. Knopf Inc.

Dodge, C. & Jerse, T. 1997. *Computer Music : Synthesis, Composition, and Performance*. Second Edition. New York: Schirmer Books.

Eimert, H. 1955. "What is electronic music?" *Die Reihe* Vol. 1.

Ernst, D. 1977. *The Evolution of Electronic Music*. New York: Schirmer Books.

Harris, C.M. 1957. *Handbook of Noise Control*. McGraw-Hill Book Company.

Hoffmann, Peter. 1997. "The New GENDYN Program" *Computer Music Journal*. Cambridge, Massachusetts: The MIT Press.

Koenig, G.M. 1970. "Project 2, a programme for musical composition" *Electronic Music Report 3*. Utrecht: Institute of Sonology.

Koenig, G.M. 1971. *Summary Observations on Compositional Theory*. Utrecht: Institute of Sonology.

Lorrain, D. 1980. "A Panoply of Stochastic 'Cannons'" *Computer Music Journal* 4(1). Cambridge, Massachusetts: The MIT Press.

McAulay, R. J. and Quatieri, T. F. 1985. "Speech Analysis/Synthesis Based on a Sinusoidal Representation" *Technical Report 693*, Lincoln Laboratory, M. I. T..

Peterson, A. & Gross, E. 1963. *Handbook of Noise Measurement*. Fifth Edition. General Radio Company.

Press, W., Teukolsky, S., Vetterling, W., Flannery, B. 1993. *Numerical Recipes in C : The Art of Scientific Computing*. Second Edition. Cambridge University Press.

Roads, C. 1996. *The Computer Music Tutorial*. Cambridge, Massachusetts: The MIT Press.

Robindoré, B. 1996. "Eskhaté Ereuna: Extending the Limits of Musical Thought - Comments On and By Iannis Xenakis" *Computer Music Journal* 20(4). Cambridge, Massachusetts: The MIT Press.

Russolo, L. 1986. *The Art of Noises*. Pendragon Press.

Serra, Marie-Hélène. 1993. "Stochastic Composition and Stochastic Timbre : Gendy3 by Iannis Xenakis" *Perspectives of New Music* Vol. 31 No.1.

Serra, X. 1989. "A System for Sound Analysis/Transformation/Synthesis Based on a Deterministic Plus Stochastic Decomposition." Ph.D. dissertation. Stanford: Dept. of Music, Stanford University.

Stockhausen, K. 1959. "`Two Lectures: I. Electronic and Instrumental Music" *Die Reihe* Vol. 5.

Stockhausen, K. 1991. *Stockhausen On Music*. Compiled by Robin Maconie. Marion Boyars Publishers.

Tempelaars, S. 1996. *Signal Processing, Speech, and Music*. Lisse: Swets & Zeitlinger B.V.

Xenakis, I. 1992. *Formalized Music*. Revised edition. New York: Pendragon Press.