6 Algorithmic composition

KARLHEINZ ESSL

Dedicated to Gottfried Michael Koenig for his 80th birthday

Although Algorithmic composition became popular with the rise of computers, algorithmic thinking is far older – it can be traced back to the ancient times of Pythagoras and the Jewish Kabbalah. It is a method of perceiving an abstract model behind the sensual surface, or in turn, of constructing such a model in order to create aesthetic works. Behind the various approaches there is one common denominator: a longing to create something infinite that exceeds the limited horizon of our individual knowledge. Seen in this light, algorithmic thinking and its application in the arts can become a way to gain experience and to overcome barriers that are either implicit in ourselves, or erected by our social environment.

In this article, I am focusing exclusively on the use of algorithms in the compositional process, leaving aside other approaches like the algorithmic simulation of musical styles, the computational modelling of music cognition and the application of artificial intelligence techniques (Cope 1996; Ebcioglu 1990).[1] My primary aim is to demonstrate how the algorithmic spirit has evolved through the centuries – from medieval music theory to the interactive realtime-generated computer music of today.

Algorithm

The term algorithm was phonetically derived from the name of the Arab mathematician Muhammad ibn Musa al-Khwarizmi (ninth century) who introduced Hindu–Arabic numerals and the concepts of algebra into European mathematics. An algorithm can be defined as a predetermined set of instructions for solving a specific problem in a limited number of steps. Algorithms can range from a mere succession of simple arithmetical operations to more complex combinations of procedures, utilising more involved constructions from computer science such as rule-based grammars, recursion and probabilistic inference.

Although the term stems from mathematics and natural sciences its application nowadays is widely used in the field of art production and composition, especially in the domains of media art and computer music. The

'Algorithmic Revolution' in the arts started as far back as the 1950s and has drastically changed not only the way in which art is produced, but also the function and self-conception of its creators.[2] With the help of algorithms, the composer is no longer a demiurge who controls every tiny detail of a composition through the power of his imagination. By utilising algorithmic methods such as automatisms, random operations, rule-based systems and autopoietic strategies, some artistic decisions are partly delegated to an external instance. This might be regarded as a weakness of the subjective autonomy. On the other hand, it enables one to gain new dimensions that expand investigation beyond a limited personal horizon. From this basis, algorithms can also be regarded as a powerful means to extend our experience – they might even develop into something that may be conceived as an 'inspiration machine'.

Within the field of algorithmic composition the algorithm constitutes an abstract model which defines and controls some or all structural aspects of the music. This model can also serve as a generator that is capable of producing the piece as a possible variant within a field of possibilities. The latter approach can be implemented as a computer program, but the underlying idea is much older. In 1787, Johann Wolfgang von Goethe wrote a letter to his friend Johann Gottfried von Herder in which he reported enthusiastically about a discovery that he made in the botanic garden of Palermo, Sicily:

> The primordial plant would be the most wonderful creation of the world, for which nature itself should envy me. With this model and the key that it contains, one could invent an infinite number of plants, ones that despite their imaginary existence could possibly be real, thus which are not solely literary and painterly shadows and illusions, but which possess an inner truth and necessity. This same principle would be applicable to every other aspect of life as well. (Goethe 1787)

Goethe's primordial plant does not exist in reality. It is an abstract model from which every plant – and even those which do not exist! – can be deduced. A concrete form (for example, a beech tree) can be described as a structural variant of the general 'tree' model. This concept is also applicable to music – a specific model that describes a compositional 'gestalt' would allow for the generation of thousands of different variants.

Due to its rule-based nature, every algorithm can be expressed as a computer program. However, the use of algorithms is not solely restricted to computers, as we will see in the following section where I will briefly outline the historical development of algorithmic strategies in music and literature.

History

Rule-based and semi-automatic compositional methods are not at all an invention of our modern era – one can trace these principles back to the very beginning of European polyphony in the Middle Ages. In *Musica enchiriadis* (*c.* 895), Hucbald of St Amande described a method for improvising a second voice to a given Gregorian chant by singing in parallel intervals such as fourths and fifths – a practice which was later described as Organum. Instructions like these were called *canon* (from the Greek word *kanon* = rule) and had their first bloom in the Franco-Flemish polyphony of the fifteenth century. The prevailing method was to write out a single voice part and to give instructions as to how additional voices could be derived from it.

A highly complex form of the so-called riddle canon written by Johann Sebastian Bach entitled *Verschiedene Canones über die ersten acht Fundamental-Noten vorheriger Arie von J. S. Bach* (Various canons on the first eight fundamental notes of the previous aria by J. S. Bach, BWV 1087) was discovered in 1974 – a single handwritten leaflet containing fourteen different canons based on the ground of Bach's *Goldberg Variations* (BWV 988). In this piece, Bach supplies a highly compressed code, but without the algorithm that expands the rudimentary notation into something resembling a score. As this work constitutes a musical riddle, the solution has to be found by the user, through combining the various subjects in different forms such as retrograde, inversion and the retrograde of the inversion, partly with changing temporal compression or augmentation. Reinhard Böß was able to discover all possible solutions – 269 movements providing seventy minutes of astonishing music (Böß 1996).

These predecessors demonstrate the rule-based aspects of algorithmic composition. More or less automatically they yield stunning musical results which can be seen as the unfolding of the inherent potential of the material supplied.

A promising extension was the employment of random decisions in order to create music which is not limited to a fixed appearance. The invention of musical dice games by composers like Johann Philipp Kirnberger, Maximilian Stadler and Joseph Haydn[3] enabled musical amateurs to generate numerous variants of dance pieces. One of those musical construction sets is attributed to Wolfgang Amadeus Mozart. In *Musikalisches Würfelspiel* (Musical dice game, KV Anh. 294d, 1787), eleven different versions of each bar of the minuet have been composed beforehand. These bars can be linked together in any combination. By throwing two dice[4] (the sum of which provides eleven numbers from 2 to 12), one can create numerous variations of the underlying metric and harmonic model (Prieberg 1960).

As an exemplar of the manifold techniques of algorithmically inspired music particularly investigated in the twentieth century, and to be discussed further later in this chapter, one sometimes overlooked composer shall be mentioned here. Combinatorics coupled with a severe refusal to express personal sentiment characterises Josef Matthias Hauer's late *Zwölftonspiele* (Twelve-tone games) which he composed from 1940 to 1959. By applying his 'Klangreihen' algorithm which automatically generates a sequence of twelve four-part chords from any twelve-tone row, Hauer obtains sonorous chord progressions which serve as a harmonic model for a piece. The act of composing is reduced to building figurations and arpeggios on this 'ground'. For this, Hauer often used well-known baroque models, as he was not interested in inventing music himself – he rather considered himself a medium that transformed cosmic vibrations into sound.

Algorithms that are based on combinatorics and permutations, however, have a much longer history and are deeply rooted in kabbalistic thinking as it is expressed in the book *Sefer Yetzirah* (Book of Creation). This school of thinking probably influenced the Catalan monk Ramon Lull (1235–1315): in *Ars Magna* (Great art, 1305) he describes a method of combining attributes that describe the properties of God. Like in Mozart's dice game, a fixed matrix, now filled with carefully selected words, contains the basic material. The permutation is carried out by a machine (called the 'Lullian Circle') which consists of several rotating paper discs inscribed with symbols that refer to the divine attributes. By rotating these discs numerous permutations of the basic attributes could be created, generating universal conclusions about God (Eco 1995).

In the seventeenth century, the German poet Georg Philipp Harsdörffer systematically transferred Lull's concept to linguistics and literature. An algorithm described in his book *Fünffacher Denckring der teutschen Sprache* (Five-fold thought ring of the German language, 1636) demonstrates how anyone can generate all existing and potential German words through the combination of basic syllables (Cramer 2005).

Using formalisms in order to challenge the writer's and reader's imagination is the core idea of *Oulipo* ('Ouvroir de littérature potentielle', which translates roughly as 'workshop of potential literature') – a group of mainly French writers and mathematicians founded in 1960 in Paris. One of its members was Georges Perec (1936–82), well-known for setting up a system of constraints which is used as a means of triggering ideas and inspiration, most notably in his masterpiece *La Vie mode d'emploi* (Life: a user's manual, 1978). By employing a magic square he created a complex system which generated a list of items, references or objects that each chapter should allude to. Perec was also interested in computers and their algorithms, which stimulated him to create the radio play *Die Maschine* (The machine) – here we can literally hear how a computer program analyses Goethe's famous poem *Wanderers Nachtlied* (Wanderer's night song, 1780), and how it later on generates bizarre variants by applying various text-transforming algorithms to this basic material (Perec and Klippert 1972).

Pioneers

After this condensed trip into the history of algorithmic thinking, I want to introduce a pioneer in this field who is nearly forgotten nowadays. The Ukrainian composer Joseph Schillinger (1895–1943) emigrated to the United States in 1928 where he became a highly influential teacher and theorist. With the *Schillinger System of Musical Composition* (published posthumously in 1946) he developed a method of musical composition based on mathematical processes and algorithms, long before composers like Xenakis and others. Many of his concepts have penetrated modern compositional practice, from Allen Forte's work on pitch-class sets, to Karlheinz Stockhausen's so-called 'Formant-Rhythmik' or Gottfried Michael Koenig's concept of periodicity as it is implemented in his algorithmic composition software *Projekt 1*.

In the preface of Schillinger's book, Henry Cowell wrote:

> The idea behind the Schillinger System is simple and inevitable: it undertakes the application of mathematical logic to all the materials of music and to their functions, so that the student may know the unifying principles behind these functions, may grasp the method of analyzing and synthesizing any musical materials that he may find anywhere or may discover for himself, and may perceive how to develop new materials as he feels the need for them. Thus the Schillinger System offers possibilities, not limitations; it is a positive, not a negative approach to the choice of musical materials. Because of the universality of the aesthetic concepts underlying it, the System applies equally to old and new styles in music and to 'popular' and 'serious' composition.

In a highly systematic way, Schillinger invented algorithms for generating or transforming melodies, rhythms and musical forms: techniques that can be considered as tools for artistic imagination. Moreover, he applied his concepts also to graphic design (Schillinger 1948, p. 419) and even to colours (p. 346).

In 1955, the first experiments with computer-generated music were conducted by the chemistry professor and trained composer Lejaren A. Hiller (1924–94) together with Leonard M. Isaacson at the University of Illinois. They applied probabilistic algorithms – that were used for the calculation of polymer configurations – to music. The composition process was carried out by a three-step approach: a 'generator' which often employs random operations supplies the basic material; a 'modifier' applies transformations on it; and finally a 'selector' filters out unwanted results by testing the output against a rule-based system (Hiller 1959).

In 1957, the first complete computer composition – the well-known *Illiac Suite* for string quartet in four movements – was created, calculated by the University of Illinois's ILLIACI (Illinois Automatic Computer). The output was transformed manually into musical notation and performed by human musicians. Each of the four movements was carried out as a particular musical 'experiment', based on random notes controlled by rules of sixteenth-century counterpoint, paradigms of twelve-tone music or probability operations such as Markov chains.[5]

In the late 1950s and early 1960s, Hiller's experiments finally led to the development of MUSICOMP, one of the first computer languages for computer-assisted composition which basically consists of a library of subroutines. Afterwards, he also collaborated with John Cage on the multimedia piece *HPSCHD* (to be discussed later) and expanded his score-generating algorithms by digital sound synthesis (Hiller 1981).

Serialism

The Second World War not only destroyed cities and landscapes in Europe but almost eradicated musical life. Composers like Schoenberg had to flee from the Nazis, leaving a vacuum which was hard to fill. After the war, young composers from different parts of Europe – Luigi Nono, Karlheinz Stockhausen, Pierre Boulez (to name only a few) – gathered together at the Darmstadt summer courses where they tried to reconnect to the musical avant-garde which had nearly been extinguished by the Hitler regime. Confronted with the situation of a 'tabula rasa' they strived towards a 'musica pura' by developing a new musical grammar free of historical or traditional references.

For this bold endeavour a starting point was found in the late works of the Austrian composer Anton Webern (1883–1945) who, to composers such as Stockhausen and Boulez, seemed to have extended the serial principles from the mere determination of pitches to other aspects of composition such as the organisation of time, timbre and dynamics. Although it is widely accepted nowadays that this was not entirely true of Webern (who was still rooted in the classic-romantic tradition which he compressed into highly expressive micro-gestures), this productive misconception served as a trigger of enormous influence, even until today.

In Darmstadt serialism, Schoenberg's dodecaphonic technique, which predetermines all pitch material, is extended to the other so-called 'parameters' of musical structure like duration, dynamics and timbre. The series becomes a unifying principle which can control every detail of a composition – it affects all aspects of a musical structure, comparable with the DNA of a biological cell.

What characterises a series? A set of values from a scale with equal steps which are arranged in a specific order. It is much more than a thematic invention as it was once considered by Schoenberg: the series serves as the basic organisational principle and the core algorithm for a whole composition.

Numerous variations of the primary row can be obtained by applying transformations such as transposition, inversion, retrograde, and permutation. In order to achieve this by mathematical operations, it is necessary to translate the elements of a tone row first into a numerical representation.

A twelve-tone row

E♭ – D – A – A♭ – G – F – E – C# – C – B♭ – F – B

can be transformed into a sequence of numbers (where we define C as index 0):

3 2 9 8 7 6 4 1 0 10 5 11

By using this number list, we can easily apply a basic transformation algorithm such as transposition. If we want to transpose this row a fifth up, we just need to add the number 7 (fifth = 7 half-tone steps) to each element of the list:

10 9 16 15 14 13 11 8 7 17 12 18

and perform a [modulo 12] operation afterwards in order to map the result back into the range 0–11.

10 9 4 3 2 1 11 8 7 5 0 6

As the series is now represented as a list of numbers, we can employ mathematical algorithms in order to create variations from its primary form. If we consider the numbers as addresses, it is even possible to utilise a row as its own permutation program.

Each number in the series can be seen as being contained in an indexed slot.

series	10	9	4	3	2	1	11	8	7	5	0	6
index	0	1	2	3	4	5	6	7	8	9	10	11

If we now decide that each number of the series will represent an index, then we can derive a new series by looking up the number contained at that index. In this case, the number 10, which in the original series is in the first position, by becoming an index, points to 0, as that is the number contained in the tenth position of the original series. In this manner we arrive at the following permutation:

series 0 5 2 3 4 9 6 7 8 1 10 11

This result in turn can be used iteratively for generating further permutations, and so forth . . .

Furthermore, we can use the numbers of the series as pointers to values of another list, e.g. of dynamic values. Applying the transposed row from above onto an array of twelve dynamic values from *pppp* to *ffff*,

pppp	*ppp*	*pp*	*p*	*quasi p*	*mp*	*mf*	*quasi f*	*f*	*ff*	*fff*	*ffff*
0	1	2	3	4	5	6	7	8	9	10	11

would yield the following sequence of dynamics:

fff ff quasi p p pp ppp ffff f quasi f mp pppp mf

In his *Structure 1a*, Pierre Boulez extended this principle to the organisation of rhythmical values and articulations. In one single night in 1951, Boulez composed this piece for two pianos subtitled 'A la limite du pays fertile' (At the border of the fertile country), named after the watercolour painting *Monument im Fruchtland* (Monument in the fertile country, 1929) by Paul Klee. This picture shows an abstract composition of orthogonal shapes, a pattern which brings to mind a rural landscape seen from an airplane. Boulez referred to the title as a metaphor for an attempt to test the serial method by developing a strict algorithm which was designed to eliminate subjectivity and personal taste.

Structure 1a is entirely based on a twelve-tone row (in fact the same series we used earlier) which Boulez extracted from Olivier Messiaen's piano piece *Mode de valeurs et d'intensités*, a piece that Messiaen wrote while teaching at the Darmstadt summer course in 1949 where he was extending the serial principle to the organisation of rhythm and dynamics. As György Ligeti describes in his analysis of *Structure 1a* (Ligeti 1958), Boulez used all twelve transpositions of Messiaen's series in four different forms – original, retrograde, inversion and the retrograde of the inversion – represented as a matrix where the note names have been replaced by numbers (as shown in the example above). This representation allows the mapping of rhythmical values to numbers which were obtained from the original dodecaphonic series; by multiplying the numbers from a series with the basic rhythmical unit – the demisemiquaver – Boulez creates series of rhythmical durations.[6]

The same is done with the dynamics, as described earlier in this chapter, and also – partly – with articulations.

This automatic process is carried out by a simple mapping algorithm that generates independent series for pitch classes, durations and dynamics. By superimposition, the parameter series are combined into a linear musical structure which manifests itself as a monophonic melody. These 'melodies' are then woven together in a polyphonic sense, forming a fabric of piano sounds.[7]

It is important to understand that by composing a piece with strictly predetermined material drawn from an automatism, many artistic decisions are replaced by an algorithm. Although highly ordered by predetermination, the result appears as statistical agglomeration of points in space and time.

These experiments were important to the development of algorithmic composition as they shifted compositional thinking into the domain of algorithms. Furthermore, they provided the conceptual basis for the first computer programs which generated musical structures, such as Gottfried Michael Koenig's *Projekt 1* (1963).

Stochastic music

The fact that the strict predetermination of serialism results in an incomprehensible auditory chaos was fiercely criticised by Iannis Xenakis in 'The Crisis of Serial Music':

> Linear polyphony is self-destructive in its current complexity. In reality, what one hears is a bunch of notes in various registers. The enormous complexity prevents one from following the tangled lines and its macroscopic effect is one of unreasonable and gratuitous dispersion of sounds over the whole sound spectrum. Consequently, there is a contradiction between the linear polyphonic system and the audible result, which is a surface, a mass. (Xenakis 1955)

Xenakis proposed a solution to this problem through the use of statistical methods in musical composition:

> In fact, since these linear combinations and their polyphonic superpositions are no longer workable, what will count will be the statistical average of isolated states of the components' transformations at any given moment . . . Hence, the notion of probability is introduced, which, by the way, implies combinatory calculus in this specific case. (Xenakis 1955)

Xenakis suggested replacing the deterministic causality of serialism with the more general concept of probabilistic logic which could contain the strict serial approach as a particular case.

What Xenakis calls 'stochastic music' is based on random operations within time-variable constraints. The principle of indeterminacy and the statistical organisation of mass structures can also be found in nature – 'natural events such as the collision of hail or rain with hard surfaces, or the song of cicadas in a summer field' (Xenakis 1961). This concept subsequently led to Granular Synthesis which later became a core aspect of Iannis Xenakis's graphical UPIC system, a computer-based machine dedicated to interactive composition.

Xenakis first applied this concept in the 1950s to generate scores for instrumental music, mostly large orchestral works which employ a lot of musicians. In his orchestral piece *Metastasis* (1953–4) for sixty-one players, the single instruments cannot be perceived as individual voices as they melt together into a collective sound field. The mass structures of this piece, however, are not calculated by statistical methods – here Xenakis uses graphical methods which later led him to formulate the architectural design for Le Corbusier's Philips Pavilion for the Brussels World's Fair in 1958 (Xenakis 1961). *Pithoprakta* (1955–6) was the first work which he composed employing statistical methods and was based on Maxwell-Boltzmann's kinetic theory of gases, simulating the Brownian motion of gas molecules ricocheting off each other.

Having used composition algorithms without the aid of computers since the 1950s, Xenakis' probabilistic concepts could finally be implemented as computer programs using the FORTRAN programming language, which was running on an IBM-7090 mainframe computer in Paris (Xenakis 1965). This led to a series of works entitled *ST-xxx* (where ST stands for stochastic) composed in 1962, which includes pieces for string quartet, various ensembles and orchestra. Xenakis first designed flow charts for the various subroutines which he afterwards translated into computer language.[8] The result of the calculation was printed as a score list which later had to be transformed into musical notation.

The use of the computer was a relief for it freed the composer from painstaking bookkeeping tasks and shifted his attention to the exploration of the unknown:

> Freed from tedious calculations, the composer is able to devote himself to the general problems that the new musical form poses and to explore the nooks and crannies of this form while modifying the values of the input data. For example, he may test all instrumental combinations from soloists

> to chamber orchestras, to large orchestras. With the aid of electronic computers the composer becomes a sort of pilot: he presses the buttons, introduces coordinates, and supervises the controls of a cosmic vessel sailing in the space of sound, across sonic constellations and galaxies that he could formerly glimpse only as a distant dream. (Xenakis 1965)

Field composition/aleatoric composition

The pointillistic fragmentation caused by early serialism was not observed solely by Xenakis, but also by the serial composers. However, this reflection did not lead to the abolition, but to the refinement of the original simplistic concept. By extending the determination of single elements to several parameter values at once, Stockhausen introduced the notion of a 'group' where the parameters do not constantly change, being maintained for a longer time. He also formulated higher-level parameters like 'density' which control qualitative musical attributes on a global level. This was exemplified in his *Gruppen* (1957–9) for three orchestral groups and was accompanied by a thorough theory of musical time, formulated in his excellent article '. . .how time passes . . .' (Stockhausen 1957).

By this time, the former pointillistic serialism had mutated into so-called 'field composition' which provided methods for creating musical structures that were capable of building musical forms again. A 'field' creates a synthesis between serial determinism and chance composition, as the single element is defined within a global context which described a 'field of possibilities'.

Paradoxically, in order to gain more control over the musical structure it had become necessary to involve random principles. Unlike Xenakis, who employed stochastic methods like the Gaussian distribution or Markov chains, serial composers – namely Gottfried Michael Koenig – have replaced the serial permutation mechanism with a non-deterministic, but promising strategy – the use of aleatoric principles.[9]

Gottfried Michael Koenig used this method for the composition of his *Streichquartett 1959* for which he developed a powerful algorithm defining a flexible multi-level process which involves random operations. Starting from a 'supply' which comprises all values of a certain parameter – for instance a list of possible rhythmic values – he obtains a data field by randomly selecting from the supplied elements. By repeating those elements according to a group list (which is also calculated by chance operations), a repetition list is created which contains the elements of the selection in different weights. Finally, this repetition list is brought into a random order and represents the result which can – for instance – be translated into a rhythmical structure (Essl 1989):

```
material supply     [a b c d e f g]

                         |
                         | random selection
                         |
                         v

selection           [a c d g]       [3 1 2 5]      group list

                         |                  |
                         | element          |
                         | repetition       |
                         |                  |
                         v                  v

repetition list     [a a a c d d g g g g g]

                                 |
                                 | random order
                                 |
                                 v

result               [c d g g a g g d a g a]
```

Although the entire work was composed by employing random decisions, it does not sound like arbitrary chance music thanks to the intelligence and skill that was put into the definition of the composition algorithm. However, at this time Koenig did not use a computer program – he had to do everything by hand (Essl 1989).

In 1963, Gottfried Michael Koenig began work on a composition[10] that was based on an algorithmic model which he implemented as a computer program – *Projekt 1* (PR1). Instead of rows, he assembled lists of parameter values, and in place of deterministic permutation algorithms he used random operations which select values for the four prototypic parameters of pitch, rhythm, dynamics, timbre. The random decisions were controlled by a group mechanism which determined its variety in time. Thus, Koenig was able to create transitions between regular and irregular events.[11] Moreover, he encoded the dialectic between determination and chance into an algorithm which could map various states between those antipodes (Koenig 1979).

> I had the idea of collating my experience with programmed music at the desk and in the electronic studio to form a model which would be able to produce a large number of variants of itself almost fully automatically. Faithful to the fundamentals of the nineteen-fifties, all the parameters involved were supposed to have at least one common characteristic; for this I chose the pair of terms, 'regular/irregular'. 'Regular' means here that a

> selected parameter value is frequently repeated: this results in groups with similar rhythms, octave registers or loudness, similar harmonic structure or similar sonorities. The duration of such groups is different in all parameters, resulting in over-lappings. – 'Irregularity' means that a selected parameter value cannot be repeated until all or at least many values of this parameter have had a turn. The choice of parameter values and group quantities was left to chance, as was the question of the place a given parameter should occupy in the range between regularity and irregularity.
> (Koenig 1978)

PR1 in its original form was a computer program that did not allow any user interaction. As an hermetic system that contained the algorithms and also all necessary data (like a list of rhythmic values, a defined number of dynamics and instruments) it would produce nearly infinite variants of the same structural model which it output as a score list. This limitation, however, was removed during the further development of the program and by virtue of the fact that it was also used by other composers. Still, the degree of determining the input data is limited: 'A composer using this program only has to fix metronome tempi, rhythmic values and the length of the composition: in other words, he only decides on the time framework of the result, and this only roughly, because all details are generated by the automatism of the program' (Koenig 1980).

The output of PR1 is a score list comprised of six columns: line number, instrument, entry delay (ED), pitch, register, and dynamics. Each row defines a musical event:[12]

#	Instr	ED	Pitch	Register	Dynamics
1	8	1/8	F	5	*mf*
2	8	1/8	D	4	*fff*
3	1	1/8	C#	6	*ff*
4	9	1/8	A# G#	3 4	*ppp*
5	3	1/8	G	6	*p*
6	6	1/8	F D	5 5	*f*
7	6	1/8	C#	3	*pp*
8	6	1/8	A#	3	*pp*
9	4	1/8	G# G	6 6	*pp*
10	5	0/0	F	4	*pp*
11	7	4/5	D C# B	5 5 5	*pp*
12	7	3/4	G# G F D C#	4 4 4 4 4	*pp*

In this example, one can clearly see that the rhythm of the beginning is very regular, whereas the registers shift quite erratically.

The output as it is represented in the score table needs further refinement before it eventually becomes music, as the score list lacks some definitions – rhythm is only defined by entry delays (no rhythmic durations are supplied). Conglomerations of pitches such as in line 12 can be interpreted as straightforward chords, but might also be used to invent various modes of 'arpeggios'.

First, the score table has to be transcribed into an intermediate score ('particello') using traditional music notation. Then, after thorough analysis, the user might develop a specific strategy with which to interpret the aforementioned particello and to translate it into a composition which can be a piano piece, a string quartet or even an orchestral work. Gottfried Michael Koenig still uses *Projekt 1* nowadays. In 2005, he released a final version of his program that allows a more detailed specification of the input data.[13]

Chance

Whereas Xenakis and Koenig employ random operations within the context of an algorithmic model in order to gain control over the musical structure, John Cage uses chance decisions for achieving quite the contrary – creating music that is not defined by personal taste or individual dislikes. Instead of exploiting music for representing order systems or expressing subjective sentiments, the sounds are freed from meaning and historical connotations, free 'to come into their own' (Cage 1959).

In order to achieve this goal, Cage invented numerous algorithmic systems that employ chance operations (obtained from the Chinese oracle book 'I Ching' as well as from star atlases or by graphical methods) which he used for the selection and coordination of musical events. Despite the use of chance methods, the basic material of each piece was carefully chosen by Cage and shaped by his personal predilections. Organised in 'charts' (Pritchett 1993) elements were selected and combined by randomly controlled algorithms, with the aim of overcoming habitual modes and creating unpredictable results.

Cage's thinking is full of antagonisms and paradoxes – he adopts painstakingly elaborated control mechanism in order to lose control and to liberate music from the imperatives of human intervention. This, however, could only be achieved by algorithms that would abrogate the subjective intentions.

Cage had collaborated with Lejaren Hiller in the 1960s, who had written for him, amongst other software, a computer implementation of the 'I Ching'. Such a program enabled Cage to concentrate more heavily on the compositional outline of a piece and freed him from the time-consuming and unproductive task of tossing coins in consulting the 'I Ching'. They also collaborated on the gigantic multimedia spectacle *HPSCHD* (1967–9) which involved three sets of computer programs, one of them for composing the harpsichord part derived from Mozart's *Musical Dice Game* (Hiller 1981).

In the last decade of his life, John Cage also worked with Andrew Culver, who wrote computer programs according to Cage's instructions. This was especially helpful for mastering a work like *Music for*... (1984–7) which consists of numerous individual instrumental parts which have been generated by a computerised algorithm.

During the 1950s, while Cage was developing his concepts of chance composition and indeterminacy, he also created graphical 'tools' for musical composition. Instead of supplying a written score, a piece like *Fontana Mix* (1958) was delivered as '10 transparent sheets with points, 10 drawings having six differentiated curved lines, a graph (having 100 units horizontally, 20 vertically) and a straight line, the two last on transparent material' (Cage 1958). Together with the precise instructions, the performer can create any number of compositions.

The ingredients of the 'construction kit' have to be overlapped in random positions and placed over a page that contains six curved and intermingled lines. By measuring positions of intersecting lines, six parameter values which are needed to determine a single sound event are obtained. The choice of the parameters is free. 'Where possible technically this can be not only simple changes of time (starting, stopping) but also alterations of frequency, amplitude, use of filters and distribution of the sound in space' (Cage 1958).

Generative music

Cage's ideas of freeing the music from its social and historical bondage can be expanded to its primary obstacle – the temporal limitation. Music, whether it appears as a 'classical' composition, a traditional gamelan piece or a rock song, is defined by a beginning and an end, as opposed to natural sound phenomena like the rustling of leaves in the wind which can be regarded as an (infinite) stream of sound. Creating music that lasts forever without repeating itself became a concern of the pop artist Brian Eno, which he first realised in the sound environment *Music for Airports* (1978) for the 'Marine Air Terminal' of New York's LaGuardia Airport. This piece was created by simultaneously playing tape loops of different lengths resulting in an ever-changing, ambient-like sound texture of unpredictable combinations, an approach that was obviously inspired by some works of American

Minimalists like Steve Reich (*It's gonna rain*, 1965) and Terry Riley (*In C*, 1964). By employing a simple looping algorithm, Eno was able to create infinite soundscapes that are based only on a few basic elements. In 1990, his ideas were taken on by the software engineers Pete and Tim Cole for the development of KOAN – a commercial program for generative music in the style of Brian Eno's ambient music. With this software environment, one can define rhythmic models, harmonic progressions, selections of sounds and specify correlations and random variabilities. With such a set of rules an infinite sound stream is generated in realtime, as described by Eno: 'Ordinary music is like engineering, where everything's built according to a plan, and it's the same every time you play it. Generative music is more like gardening; you plant a seed, and it grows different every time you plant.'[14]

This statement contains an interesting accordance with Goethe's primordial plant (see above). Furthermore, Eno's ideas also relate to Erik Satie's 'furniture music' (1917) – background music which was originally played by live performers outside a concert situation. In the 1960s, this concept was rediscovered by John Cage and the American Minimalists. This finally led to the development of ambient music which is 'able to accommodate many levels of listening attention without enforcing one in particular; it must be as ignorable as it is interesting,' as described by Eno in the liner notes of *Music for Airports*.

Using generative composition algorithms on computers, music can be created in realtime by an autonomous and infinite automatic process. As this music has no beginning nor end, the distribution on a reproductive medium such as a compact disc seems highly inappropriate. This consideration led Maurice Methot and Hector LaPlante to the instantiation of 'The Algorithmic Stream'. Starting in 1997 as one of the earliest streaming audio systems on the Internet, this platform broadcasts non-repeating computer-generated sound and music live and in realtime as it is produced.[15]

One successor of this project is 'rand()%', an automated net radio station streaming realtime generative music. It serves as an independent platform for artists who can submit their own computer programs that create algorithmic music on-the-fly.[16]

Realtime

In the beginning of computer music, compositional algorithms were used 'out of time' (Xenakis 1971) for creating musical scores. In most of the pieces mentioned before – as for example those of Hiller, Xenakis, Koenig and Cage – a symbolic output in the form of a score list had to be translated into musical notation in order to be performed by musicians. However, over the last twenty years the permanent improvement of computer technology and the increase of processing speed has promoted the development of algorithmic composition environments which generate sound directly in realtime.

An important forerunner was the Institute for Sonology in Utrecht which was headed by Gottfried Michael Koenig between 1964 and 1986. Starting from compositional strategies that he had developed during the previous decade, a modular analogue studio based on the paradigm of voltage control was built. It consisted of independent hardware modules (such as oscillators, filter, envelop generators, logic circuits etc.) that could be connected with each other and the parameters of which were controlled by voltage. By creating a network of connected modules, one could implement an algorithm that produced sound in realtime.

The core of this system was a so-called 'variable function generator' – a programmable sequencer that stored a time-variant voltage function which could be read out at different speeds or scanned by a random mechanism. This function could likewise be used as a control and an audio signal, and was utilised by Koenig as the basic structural definition for an entire composition, functioning similarly to the series in serial music. This new technology enabled Koenig to compose electronic music through automatic processes, without the need of tape montage, as in the WDR studio in Cologne (Koenig 1986).

The structural paradigm of a modular analogue studio with independent units communicating by cable connections was later transferred into the domain of a computer language. This was the advent of Max[17] – a visual, object-oriented programming language, initially designed for interactive musical performance and which is also suitable for digital signal processing as well as realtime control.

Although Max had the potential to implement algorithmic music systems due to a core architecture which enabled realtime processing of data, it was lacking higher-level compositional tools. This led to the design of the *RTC-lib* (Real Time Composition Library for Max) – a software library for algorithmic composition – which I started together with Gerhard Eckel in 1992 and the development of which continues today.[18] Its software modules offer the possibility to experiment with a number of compositional techniques, such as serial procedures, permutations and controlled randomness. It comprises low-level objects for list processing and numerous random operations, but also high-level compositional tools like rhythm generators, harmony generators and envelope processors (Essl 1996).

The *RTC-lib* was utilised for the first time in *Lexikon-Sonate* (1992–2007) – an infinite interactive realtime composition environment that generates piano music in real time and plays it on a Yamaha Disklavier, reflecting the history of this genre from Johann Sebastian Bach to the avant-garde music of our times (Essl 1997). Originally conceived as an autonomous system acting on its own, it eventually developed into an instrument whose numerous compositional parameters can be controlled by attached MIDI fader boxes. This enables one to improvise highly complex piano music without even touching the keys of the instrument.

Recently, the realtime approach to algorithmic composition was expanded into the multi-user sound installation *Raumfaltung* (2003) where up to eight visitors could listen to eight different versions of the same piece at the same time – within a room installation by Beat Zoderer at the Kunstmuseum Bonn, Ramón González-Arroyo and Gerhard Eckel set up an environment based on the LISTEN technology (Eckel 2003). This environment tracks the movements of listeners who are themselves equipped with headphones. The individual tracking information is evaluated by a computer system that generates control data for a granular synthesis program written in Max/MSP. It takes into account the current position of the visitor within the environment and also the relative positions of the listeners to each other. Using this information, an individual version of the composition is rendered for each person and broadcast to their headphones, creating a virtual auditory space that intermingles the spatial layout of Zoderer's installation (González-Arroyo 2003). In this piece, algorithmic compositional strategies have arrived at a state where music is not merely the reproduction of an artistic oeuvre; it has become an integral part of a multi-aesthetic environment that extends our perception into the realm of immersion.

Conclusion

Thanks to fast and affordable personal computers together with high-level software environments (such as CSound,[19] SuperCollider,[20] ChucK,[21] OpenMusic,[22] PD and Max/MSP), the development of compositional algorithms has become an integral part in the creation of electronic music. Nowadays, composers are no longer dependent on academic computer music studios with their specific aesthetics and production methods. This situation creates a new self-conception of a composer. Instead of delegating the computational work to an external programmer, one becomes an autonomous artistic individual who combines technical skills and artistic visions. By creating independent networks over the internet, communities are organising themselves which serve as think-tanks and pools for sharing ideas and code.

To me, algorithmic music generation in realtime seems the most challenging aspect. It enables the development of art forms where the generative concept of Goethe's primordial plant can be interactively controlled during its creation. Now the compositional process is taking place 'in time' again, as in musical improvisation, but depends on an algorithmic framework created beforehand. By this, 'instruments' can be constructed which connect to our body, creating a physical extension of our mind.