

# **B E Y O N D D E C I S I O N S**

---

## **DIRECTIONS IN COMPUTER-ASSISTED MUSICAL COMPOSITION**

by

Adalberto Andrés Vidal

An essay submitted in partial  
fulfilment of the requirements for the  
degree of

Master of Philosophy in Musical Composition

University of Cambridge

February, 2006

# C O N T E N T S

---

<b>INTRODUCTION</b>	<b>1</b>
<b>1 RULE-BASED COMPOSITION</b>	<b>4</b>
1.1 FIRST STEPS: L. A. HILLER AND L. M. ISAACSON	4
1.2 A TWELVE-TONE COMPOSER: STANLEY GILL	6
1.3 JOHN ROTHGEB: THE UNFIGURED-BASS PROBLEM	8
1.4 A COMPOSER'S APPROACH: IANNIS XENAKIS	9
1.5 WHO IS THE COMPOSER?	10
<b>2 KNOWLEDGE-BASED COMPOSITION</b>	<b>12</b>
2.1 PROBABILISTIC ANALYSIS	12
2.2 COMPUTER COGNITION?	15
2.3 SYNTHESIZING STYLE: DAVID COPE	18
2.4 WHERE IS THE COMPOSER?	21
<b>3 ORGANIC COMPOSITION</b>	<b>23</b>
3.1 CELLULAR AUTOMATA	24
3.2 GENETIC ALGORITHMS	25
3.3 NEURAL NETWORKS	26
<b>4 FINAL THOUGHTS</b>	<b>28</b>
4.1 BEYOND DECISIONS	29
4.2 COMPUTER-ASSISTED MUSICAL COMPOSITION	29
<b>INDEX</b>	<b>31</b>

# I N T R O D U C T I O N

---

The process of musical composition can be characterized as involving a series of choices of musical elements from an essentially limitless variety of musical raw materials.<sup>1</sup>

Lejaren Hiller's and Leonard Isaacson's view of composition as a process of decisions is quite clear and perhaps also rather over-generalized. But it was the necessary natural starting point for them, who were two of the pioneers in the field of computer music composition and the first who experimented on the transfer of some of the decision-making processes involved in musical composition from the human composer to an electronic computer.

The study of their experiments as well as further ones raises two significant questions: is a computer really capable of taking decisions? And, also, if the beautiful sounds sometimes produced by waterfalls, trees, or any other natural sources are not considered music since they are not organized by humans, why then should we regard sounds and structures organized by a computer as being music?

These questions are closely related; if we consider, like Hiller and Isaacson, that music is a result of a decision-making process then we should exclude from its realm all sound structures not shaped by decisions. And structures generated by computers may only be judged as being music if we accept that computers are capable of taking musical decisions.

The question can be approached by making a distinction between *arbitrary* and *logical* decisions. 'Choices determined by chance, whim, or impulse, and not by necessity, reason, or principle'<sup>2</sup> are generally called arbitrary. These types of decisions, excepting perhaps the ones determined by chance, cannot be taken by computers; whim and impulse remain exclusively in the sphere of human beings.

Thus, our main question can be reformulated to whether a computer is capable of taking *logical* decisions or not. And here, we might risk an affirmative answer. However, a framework of logic must be previously defined for a computer in order to allow it to take logical decisions since, as it is also the case of the human creative process, compositional choices are not generally taken from a limitless variety of

---

<sup>1</sup> Lejaren A. Hiller, Jr. and Leonard M. Isaacson, *Experimental Music; Composing with an Electronic Computer* (New York: McGraw-Hill Book Company Inc., 1959), p. 1.

<sup>2</sup> *The American Heritage Dictionary of the English Language*, ed. by Houghton Mifflin; 4<sup>th</sup> edn., (January 15, 2000).

possibilities but rather from certain constrained universes of options, principles, and rules that might be called musical systems or styles.

It is in the initial development of these logical frameworks for computer-based musical decisions that the history of computer-assisted composition starts, and, as computers start to prove that they are able to assist the composer by taking certain musical decisions, a new role for the composer begins to appear which may be referred to as the composition of compositional systems. This new role requires a deep understanding of our own creative and cognitive process.

As 'music was predictably the art form most directly accessible to computational study',<sup>3</sup> during the early 1950s there was an initial wave of enthusiasm due to a new consciousness of the huge potential of the perspectives provided by the use of computers in musical composition. But soon, in the 1960s, computers were considered as a new instrument, and proved to be more attractive and accessible for composers if used for generating and processing audio signals rather than applied to the analysis and extension of the formal structure of music.

The main reason for this preference from the musicians' point of view was the ease of obtaining results. Composers were fascinated by the new and relatively straightforward way to obtain a universe of sound materials introduced by computers and electronic music in general, and they tended to disregard the computer's intrinsic capabilities for extending the formal structures of music.

Nevertheless, research on computer music continued to develop in several academic and specialized environments, embracing knowledge from many different disciplines such as biology, mathematics, computer science, linguistics, and psychology; in the last decade computer technology has proven to be far more accessible to musicians who are increasingly showing interest and enthusiasm in the field.

More than fifty years of research into computer music has produced an astonishing corpus of music, experiments, papers, theories, and software that ranges from an initial rule-driven computer composition to the latest investigations on artificial intelligence, cognitive science, and music.

Due to the academic environment in which these researches and experiments were carried out, there is a rich and extensive technical bibliography on the subject which in the majority of the cases comes from the same researchers and composers. Also,

---

<sup>3</sup> John Rothgeb, 'Simulating Musical Skills by Digital Computer', Proceedings of the 1979 annual conference (ACM 79), *IEEE Transactions on Computers* (1979), p. 121.

several anthologies and compendiums were published with the addition of some analytical and critical essays on them.

This essay aims to depict, through a critical and non-technical overview of the most significant directions in computer music, the emergence of a paradigm shift in the art of composing music altering the relation between the composer and the musical decisions, and significantly changing and opening up new horizons within the process of musical composition. The non-technical approach chosen for this paper is intended to help to fill the gap that exists between computer and non-computer composers.

Two recurrent questions will drive my analysis: where is the composer? Where are the decisions taken by composers?

As we shall see, human musical decisions have not been left behind but, rather, they are being extended into new domains.

# 1 R U L E - B A S E D C O M P O S I T I O N

---

Computers are capable of making logical choices based on a predefined framework of rules and conditions. Additionally, they can emulate certain arbitrariness by generating pseudo-random numbers. When combined with the intrinsic iterative nature of computer programs, these two abilities play a central part in a process in which the computer iteratively generates arbitrary or random numbers which are continuously selected or filtered according to predefined rules. The combination of random generation and logical filtering within an iterative process was the starting point for the initial experimentation carried out in the field of computer-generated music with the aim of transferring compositional abilities to computers.

The experimentation described in this chapter was principally focused on the codifications of musical rules and on the development of computational processes which allowed the computer to apply those rules with the purpose of extracting order from an initially anarchic universe of options.

There is an extensive list of researchers and composers who experimented in this topic. However, in order to stay within the scope of this essay, I will only describe some of the most significant experiments and, particularly, those that pinpointed different directions in this research field.

## 1.1 FIRST STEPS: L. A. HILLER AND L. M. ISAACSON

In 1955 Lejaren A. Hiller and Leonard M. Isaacson started to carry out the first fully-described experiments in computer-assisted musical composition. The purpose of their research consisted in the transfer of some compositional procedures to the *Illiac* computer, located at the University of Illinois, to see if a computer can produce stylistically defined music as well as novel musical structures. The output produced as a result of carrying out those experiments was used to compose a four-movement piece called the *Illiac Suite for String Quartet*.

These researchers viewed the process of creative composition as an ‘*imposition of order upon an infinite variety of possibilities*’,<sup>1</sup> and the computer seemed to be the adequate tool to be involved in the realization of that conception of music since they could be easily programmed to rapidly produce random universes of options as well as to filter groups of information from them in accordance with predefined rules. They found that composition of music could be treated by an iterative method which, in its simplest application to musical composition can be briefly summarized in two steps. In the first one, the computer generates random numbers which represent different musical parameters such as pitch, dynamics, rhythmic patterns, or articulations. Each of these random musical parameters is then evaluated, in a second step, by a series of tests expressing diverse rules of composition and either selected or rejected. If selected, the value is incorporated to the *composition*. Otherwise, a new random value is chosen and scanned. This last process is reiterated until a satisfactory value is obtained. This simple iterative method, which involves the generation of random values and their selection according to certain rules, set the basis for further experiments in the field of computer generated music, and, enriched or extended in many different ways, all further experiments in the field make use of it.

As Hiller and Isaacson were conscious that ‘computer-produced music which is meaningful is conceivable to the extent to which the laws of musical organization are codifiable’,<sup>2</sup> the central problem of their research resided, thus, in the understanding, selection, and codification of the rules of composition to be applied by the computer within the compositional process. This early researchers were very conscious of this problem, which would also be central to further investigations in the field. From the early stages of the experimentation in this subject, the task of programming a computer to compose music involved the study and examination of the means by which human beings compose and analyse music by themselves.

It is interesting to note that, when describing the aesthetic problem of their experiments, Hiller and Isaacson considered the views developed during the period of the nineteenth-century Romantic tradition to be ‘rather extreme’<sup>3</sup>. It seems that those views, which considered compositional choices as mainly driven by emotional sensitivity, did not fit with the resources of their experiments since it was impossible, or at least extremely difficult at that time, to rationalize and codify emotion or sensitivity

---

<sup>1</sup> Lejaren A. Hiller, Jr. and Leonard M. Isaacson, *Experimental Music; Composing with an Electronic Computer*, New York: McGraw-Hill Book Company Inc., (1959), p. 2.

<sup>2</sup> L. A. Hiller and L. M. Isaacson, *op. cit.*, p. 2.

<sup>3</sup> *Ibid.*, p. 11.

into rules to be applied by a computer in the compositional process. Contrastingly, they embraced what they called a more *traditional* conception of music originated by the ancient Greeks and restated among others by Stravinsky, who characterized ‘the phenomenon of music as a form of speculation in terms of sound and time’.<sup>4</sup> This speculative and formalistic conception of music encouraged the codification of rules and, thus, their first application to computer composition.

Their research was divided into four experiments. The first two were focused on the codification of strict counterpoint rules based on the formalization done by J. J. Fux<sup>5</sup> in 1725 and on the application of those rules in a computer program for generating different species of counterpoint. The third and fourth experiments were aimed to demonstrate that a computer can produce novel musical structures in a more contemporary style and to show that they can be used in highly unusual ways to produce totally different species of music. In this last experiment the researchers experimented with Markov chains and statistical procedures which will be described in the following chapter.

Hiller and Isaacson summarized the results of their experiments by indicating that computers provide new ways for studying certain techniques of musical composition and by noting that they have experimentally illustrated how various musical forms and processes of composition can be treated with the aid of computers. A performance of the *Illiac Suite* was given publicly on August 9, 1956, at a concert at the University of Illinois, and after several oral presentations of their experimental work and some extensions done to their experiments, they wrote down a complete account of their research in a book, entitled *Experimental Music; Composition with an Electronic Computer*, which was published in 1959.<sup>6</sup>

## 1.2 A TWELVE-TONE COMPOSER: STANLEY GILL

A different research instance, but related to the same view of composition as practice ‘defined in terms of extracting orderly structures out of random materials by a process of selection and rejection’,<sup>7</sup> was carried out in 1962 by Stanley Gill in response to an enquiry from the BBC to write a program for their research computer, called *Pegasus*,

---

<sup>4</sup> Igor Stravinsky, *Poetics of Music* (Cambridge, MA: Harvard University Press, 1947), p. 16.

<sup>5</sup> J. J. Fux, *Steps to Parnassus*, trans. and ed. by A. Mann, with the collaboration of J. St. Edwards, (New York: W. W. Norton & Company, Inc., 1943).

<sup>6</sup> L. A. Hiller and L. M. Isaacson, *op. cit.*

<sup>7</sup> *Ibid.*, p. 178.

with the purpose of composing music to be broadcasted in the program *Machines like Men* on Thursday, August 30, 1962.

As in Hiller's and Isaacson's experiments, the computer was used to generate a universe of options and to select among them in accord to certain rules. However, Gill's experiment was different in two aspects; instead of traditional counterpoint, the musical style and procedures codified into rules came from the technique of serial composition, and the iterative compositional method was extended into a process that was able to maintain the states of the composition in each of its steps and that might be visualized as a tree-like recursive structure.

The use of the serial technique implied a significant change in the generative aspect of the compositional process since the randomness of the initial universe of possibilities was constrained to a minimum of options derived from the different permutations of the twelve-tone row. This restriction in the amount of available options to be chosen at each of the steps of the process significantly reduced the amount of rules that were needed to be applied in order to achieve the desired musical style. This point is interesting because it manifests the difference that exists between the nature of the serial composition and non-serial or traditional composition methods. Traditional composition starts from a relatively unlimited source of possibilities, and choices are driven by cultural or artificial rules. On the other hand, the universe of initial options available for a serial composer is strictly limited to a procedure. In traditional composition decisions are bounded by what we may call styles or systems, while in serial composition choices are limited and driven by a rigorous procedure. The rules adopted for Gill's experiment were aimed at producing music in the style of Schoenberg and the main element of choice laid in the disposition of notes and rests in time.

What is most interesting about Gill's project is the tree-like process utilized in the compositional method that he transferred to the computer. This process used recursive programming and that concept would prove to be essential for further experiments in the field. The ability to maintain the various states of the composition across the different steps of the iterative process provided the means for returning to previous steps of the process without needing to restart the whole composition again. In Hiller's and Isaacson's compositional process, if at one step a satisfactory value could not be found, then the whole process of composition needed to start from the beginning. In Gill's method, as the state of the composition was maintained for each of the evaluation steps, if a satisfactory value was not found at a given step, the program would backtrack to the previous one, repeating the selection process in order to generate new alternatives.

Thus, the process was able to move back and forth until one of the branches successfully reached the number of steps intended for the composition to be complete.

In modern programming this concept can be easily implemented by means of recursive functions, but at the time of Gill's experiment it was not a trivial task and, due to the computer's limitation, only eight competitive versions of the partial composition were stored at any moment of the process.

Gill was 'relatively unmoved'<sup>8</sup> by the music generated by his program and he pointed out that the computer is hardly a substitute for the human composer but rather a new medium of expression and that perhaps, in the end, we shall see musical composition taking the form of a cooperative endeavour between the human composer and the computer.

### 1.3 JOHN ROTHGEB: THE UNFIGURED-BASS PROBLEM

Instead of using random or procedural material, as was the case of the two previously described experiments, John Rothgeb used a continuo bass as the raw material that would be used by the computer to compose harmonisations in accord to specific rules. He considered that the art of realizing an unfigured-bass is a skill that admits simulation by computer, and he codified the rules for unfigured-bass harmonisation based on the procedures set down in the eighteenth-century by Heinichen<sup>9</sup> and Saint-Lambert.<sup>10</sup>

During the seventeenth and eighteenth centuries, the keyboard player was expected to infer and supply chords and voice leading on the basis of the figures given on what was called a *figured-bass* in all music accompanied by a *basso continuo*. If the bass was unfigured, the player was still required to provide a realization of it by inferring appropriate figured-bass figures from the structure of the bass itself and by realizing it in accordance with the conventional rules.

The experiment carried out by Rothgeb in 1968<sup>11</sup> transferred that last ability to a computer by formulating as computer programs the procedures set forth by Heinichen and Saint-Lambert. What is interesting about this experiment is that, after codified and tested, the procedures were shown to be incomplete and inadequate for realizing an

---

<sup>8</sup> Stanley Gill, 'A technique for the Composition of Music in a Computer', *The Computer Journal*, London: British Computer Society, Volume 6/2, (1963).

<sup>9</sup> Johann David Heinichen, *Der Generalbass in der Composition* (Dresden: ?, 1728).

<sup>10</sup> Michel de Saint-Lambert, *Nouveau Traite de l'Accompagnement du Clavecin* (Paris: ?, 1707).

<sup>11</sup> John Rothgeb, *Harmonizing the Unfigured Bass: a Computational Study*, Ph.D. dissertation, (Yale University, 1968).

unfigured-bass. According to Rothgeb, 'it gradually became clear that general solutions to the unfigured-bass problem were probably inaccessible to procedures of the type represented by those of Heinichen and Saint-Lambert.'<sup>12</sup>

This experiment showed that the task of trying to simulate compositional skills in a computer can contribute to the study of musical theories by codifying them into computer programs and exposing their deficiencies when tested.

## 1.4 A COMPOSER'S APPROACH: IANNIS XENAKIS

The experiments previously described in this chapter were carried out with the primary purpose of investigating the possibilities and feasibility of computer creativity. In all of them, researchers codified well known styles and systems in order to provide the computer with sets of rules and conditions as a framework of logic for the decisions involved in the compositional process; none of them were entirely focused on musical results.

In the fifth chapter of his book, *Formalized Music*, and after introducing in preceding chapters some new areas of musical creation based on the theory of probability, Iannis Xenakis described how to automate through a computer the procedures that he had been using as a composer.

For a man who thought that even 'chance needs to be calculated'<sup>13</sup> the computer proved to be an ideal tool for automating and accelerating calculations. Before he addressed the challenge of automating composition by means of computers he had already developed a very strong and particular conception of musical composition in which procedures of many different types conformed the totality of a work, and this totally procedural music was naturally susceptible of being processed by a computer.

He summarizes the theoretical point of departure for the utilization of computers in composition by saying that 'the creative thought of man gives birth to mental mechanisms which, in the last analysis, are merely sets of constraints and choices'<sup>14</sup> and that, since those mechanisms can be expressed in mathematical terms, certain aspects of artistic creation may be simulated by computers.

---

<sup>12</sup> John Rothgeb, 'Simulating Musical Skills by Digital Computer', Proceedings of the 1979 annual conference (ACM 79), *IEEE Transactions on Computers*, (1979), p. 125.

<sup>13</sup> Iannis Xenakis, *Formalized Music; Thought and Mathematics in Composition* (Bloomington: Indiana University Press, 1971), p. 38.

<sup>14</sup> Iannis Xenakis, *op. cit.*, p. 133.

To carry out his experiment, he worked with two computer experts who allowed him to focus entirely on the rationalization of schemes and rules to be applied by the computer in the compositional process. The excerpt of the first flowchart of his work reflects clearly that he used the same iterative and conditional process previously used by Hiller and Isaacson as well as by the other researchers, and it is significant that the first instruction of the process was written with the following words: ‘take random note’.<sup>15</sup>

Xenakis’ experiment did not add much value or possibilities to the processes already experimented by other investigators. However, it was the first one entirely aimed and focused on new aesthetical ideas and results. Previous experiments codified well-known styles and systems; Xenakis, in contrast, codified his original ideas and thoughts about music. Thus, his experiment is very important in the sense that it signifies the first approach of a composer to the computer and because it showed that the computer can assist not only scientific researches in music, but also artistic ones.

He saw at the end of his experiment that one of the advantages of using computers for musical composition consists in that the long and laborious calculation he had had to make by hand was reduced to nothing, and that freed from tedious minutiae the composer is able to devote himself to the general problems of musical structure. He also realized that the program code itself can be seen as an objective manifestation of the musical form, a similar role to that of scores in traditional music.

The work produced by the computer was named *ST/10-1,080262*, and was performed by ten instrument players of the *Ensemble de Musique Contemporaine de Paris* at the headquarters of IBM-France on 24 May 1962.

## 1.5 WHO IS THE COMPOSER?

In this chapter I described four instances of the initial experimentation carried out in the field of computer assisted composition. All of them shared the same view of composition as a process implying the extraction of order from a chaotic universe of possibilities and, thus, they all used the same basic process of iterative random generation and logical selection. Hiller and Isaacson intended to simulate the composing process itself with computers while Xenakis used them as an aid to composition.

These experiments showed that computers could participate, to some extent, in the process of musical composition by taking care of some of the decisions traditionally

---

<sup>15</sup> *Ibid.*, p. 135.

taken by composers. Particularly, computers were very useful at the moment of taking logical decisions driven by predefined set of rules. Even though computers could participate in the compositional process, they did not replace the human composer. In fact, this initial use of computer for musical composition introduced a new role for the composer. And that role was, precisely, the creation, definition, and codification of musical rules. That role was casually, or perhaps causally, aligned with the conceptions of musical composition mostly developed in Europe after the conclusion of World War II, in which the composer was not seen as someone submissive to predefined systems or styles but, rather, as someone who creates his own systems or universes of rules.

## 2 KNOWLEDGE - BASED COMPOSITION

---

Within the experiments described above the core of the problem resided in the understanding and codification of the rules needed as a framework for the computer's operations. Soon, since the end of the 1950s, researchers and composers began to realize that that same task of analyzing and codifying musical styles and rules could also be transferred to the computer, and this new ability intended to be conveyed to the computer, required the understanding not only of certain decisions implicit within the way in which humans compose music but also of the decisions implicit in the analysis of music and how that analysis can be used to influence their compositional process.

The research into the feasibility of a computer-inductive reasoning initiated the application of artificial intelligence in music and also established the initial collaboration between different fields such as music, cognitive science, and linguistics.

### 2.1 PROBABILISTIC ANALYSIS

One of the earliest attempts to 'synthesize musical compositions by having the machine inductively analyze a sample of *acceptable* compositions and, using its conclusions, deductively synthesize new but originals compositions'<sup>1</sup> was carried out by a group of researchers led by F.P. Brooks, Jr., A. L. Hopkins, Jr., P.G. Neumann, and W. V. Wright at the Harvard Computation Laboratory in 1957.

The novelty of this experiment consisted in that almost no musical rules were *imposed* on the computer. Instead, a new process, in which the computer was able generalize rules by means of the analysis of musical samples, was integrated with an iterative compositional method to generate new musical samples related to those analysed. This integrated process was described by these researchers in what they called the *Analysis-Synthesis* theory. According to this theory, the compositional process is divided into three main stages: analysis, generalization, and synthesis. Succinctly explained, this theory stated that through the analysis of certain samples of music, a generalization is susceptible to be done, which then might be used to infer rules which are applicable to the synthesis or generation of new samples. The music samples

---

<sup>1</sup> F. P. Brooks, Jr., A. L. Hopkins, Jr., P. G. Neumann, and W. V. Wright, 'An Experiment in Musical Composition', *IRE Transactions on Electronic Computers*, EC-6 (1957), 175-182, p. 175.

generated at the end of the process should hypothetically belong to the same *class* of music from which they were derived.

In accord with that theory, there were certain variables that might affect the process; if the musical samples did not belong to the same class of music, it was unlikely that the generated music will be coherent with a particular class. On the other hand, if the samples were exceedingly alike, it was likely that the resulting music would be a replica of one of the instances of the initial sample. Also, if the samples were very small it was probable that, regardless the sophistication of the analysis, a generalization could not be drawn. The experiment that these researchers carried out had the aim of proving the statements of the *Analysis-synthesis* theory by verifying the effectiveness of the application of those three stages into an automated compositional process programmed in a computer.

Contrastingly to the rule-centred experimentation carried out by the researches described in the previous chapter, this experiment was focused on the analytical process and its codification. However, they did not try to study and replicate the mode by which humans analyse music; instead, they used a probabilistic method called *Markov analysis* named after a Russian mathematician called Andrei Andreyevich Markov.<sup>2</sup> Simply explained, this method consists in the calculation of the probabilities of the occurrences of elements in a sample. Consider, for instance, that one sample is represented by the following ten sequences of notes:

D, A, F, D, C#, D, E, F  
 D, A, F, D, C#, E, D, D  
 D, A, F, D, C#, D, F, E  
 D, A, F, D, C#, G, B, E  
 D, A, F, D, C#, D, E, D  
 D, A, F, D, C#, D, A, F  
 D, A, F, D, C#, D, E, F  
 D, A, F, D, C#, E, A, D  
 D, A, F, D, C#, A, D, D  
 D, A, F, D, C#, A, E, E

---

<sup>2</sup> Andrey Andreyevich Markov. Russian mathematician who helped to develop the theory of stochastic processes, especially those called Markov chains. Based on the study of the probability of mutually dependent events, his work has been developed and widely applied in the biological and social sciences. Retrieved February 10, 2006, from *Encyclopædia Britannica* Premium Service.

If we calculate, according to the previous sequences, the probabilities of occurrences of certain notes after C#, we shall arrive to the following percentages:

D = 50%

E = 20%

A = 20%

G = 10%

Based on those calculated probabilities it is possible to construct the following list of notes:

D, E, G, D, A, D, D, E, A, D

If then the computer is programmed to randomly select one note from that collection, a D is more likely to be chosen than a G.

The same calculation of probabilities previously exemplified for only one element of the sample can also be performed for each of the remaining elements, and that calculation may result in a complex table of probabilities that then could be applied to the generation of new samples. This whole process was naturally susceptible of being digitalized and, thus, transferred to the computer. Brooks, Hopkins, Neumann and Wright wrote a program which allowed the computer to perform the described analysis and the subsequent re-synthesis of new music. In their experiment, a sample consisted of thirty-seven common-meter hymn tunes which were chosen for the sake of simplicity. Once the hymns were chosen, the researchers encoded their rhythms and pitches as numbers and, then, they let the computer perform the probabilistic analysis along with the generation of new *synthetic* hymns. That synthesis was repeated in order to produce 600 complete new hymns based on the analysed sample. The results led the researchers to write that ‘the wide discrepancy between the ideal situation and that which currently prevails emphasizes the large amount of theoretical and experimental work that will have to be done before the inductive-deductive processes are well enough understood for general use in computing machine applications.’<sup>3</sup>

---

<sup>3</sup> F. P. Brooks, Jr., A. L. Hopkins, Jr., P. G. Neumann, and W. V. Wright, *op. cit.*, p. 182.

## 2.2 COMPUTER COGNITION?

We seem to be without a sufficiently well-defined *theory* of music that could provide that logically consistent set of relationships between the elements which is necessary in order to program, and thus to specify, a meaningful substitute for our own cognitive processes.<sup>4</sup>

That large amount of theoretical and experimental work, foreseen by Brooks et al. as necessary to understand the inductive-deductive processes involved in musical analysis and composition, did indeed happen during the years following their research. The problem that they envisioned is today one of the most vibrant fields in computer music studies. However, unlike the research of Brooks et al., succeeding research focused on the study of the human cognitive process in relation to music rather than on the development of probabilistic techniques of automated musical analysis. Practical experimentation on computer composition was, to some extent, necessarily put on hold until these new cognitive studies were able to supply theoretical frameworks capable of being formalized and codified in a computer.

In order to transfer the human's cognitive processes to a computer, researchers needed to understand fully how this cognitive process works, and this non-trivial problem is still challenging investigators at the present time. Immediately after the earlier experimentations in computer music, many researchers, composers, scientists, and musical theorists, started to develop varied approaches towards this problem. Some of these early investigations took elements from other fields, particularly from linguistics which inspired the idea of a generative grammar as a way of expressing the rules that govern most types of music. This grammatical approach represents a refinement of the rule-based automated composition described in the first chapter of this paper since generative grammar can be considered as the 'formulation of a system, or a set of rules, capable of rewriting the sequence of mental representations that are assumed to underlie the execution of activities called musical'.<sup>5</sup> However, in spite of being a refined set of rules which help to construct music, musical grammars can be used also as a mechanism for musical analysis. Otto Laske, a theorist in the field of computational analysis, specifies that there are two kinds of musical grammar: *generative grammar* and *analytic grammar*. According to Laske, *generative grammar*

---

<sup>4</sup> Barry Vercoe, 'Harry B. Lincoln, The Computer and Music, and Barry S. Brook, Musicology and the Computer', *Perspectives of New Music*, Vol. 9, N. 2, Spring – Summer, (1971), p. 323.

<sup>5</sup> Otto E. Laske, 'In Search of a Generative Grammar for Music', *Perspectives of New Music*, Vol. 12, N. 1/2, Autumn, 1973 – Summer, (1974), p. 153.

is intrinsically programmable; however, the same can not be said of *analytic grammar* since ‘the grammaticalness of musical structures is essentially inexhaustible’,<sup>6</sup> meaning that there is always certain ambiguity in the way in which analysis can be carried out, which impedes an absolute codification.

Terry Winograd, basing his research on the linguistic theories developed by Noam Chomsky, developed in 1968 a system for the automated analysis of tonal harmony which was done partly at the Massachusetts Institute of Technology and partly at the University of London. According to Winograd, ‘the problem of explaining a composition in terms of its harmonic structure is closely related to the parsing of a sentence in a language’<sup>7</sup> and, having that principle in mind, he made a linguistic analysis of tonal harmony which then he transferred to the computer in order to perform the harmonic analysis of a number of given compositions. The output of the program included ‘some quite sophisticated parsing, including the recognition of implied tonalities, modulations, and complex tonality structures’.<sup>8</sup> Winograd’s work was one of the first experiments in computational analysis of music harmony.

In 1976, Johan Sundberg and Bjorn Lindblom, influenced also by Chomsky’s work, proposed to apply the same linguistic concepts to the formalization of melody. They took for their analysis a set of Swedish nursery tunes, and they applied a rule-based generative system to produce tunes that, in their own words, were ‘felt to be similar to the melodies studied’.<sup>9</sup>

These grammatical approaches to musical analysis, among many others that for reasons of scope are not included in this paper, introduced a hierarchical conception about musical structures. Music was not better understood as being the result of the application of arbitrary rules to certain given or random material; it was beginning to be conceived as a language susceptible of being grammatically analysed. And this grammatical analysis implied the identification of structures which appeared to be hierarchically organized. An excellent description of this conception of hierarchical musical analysis was written by Fred Lerdahl and Ray Jackendoff in their *A Generative Theory of Tonal Music* (1983), where they proposed to justify that ‘a piece of music is a mentally constructed entity, of which scores and performances are partial

---

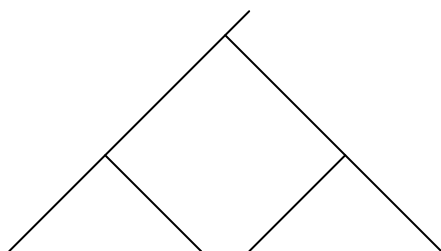
<sup>6</sup> Otto E. Laske, *op. cit.*, pp. 358-359.

<sup>7</sup> Terry Winograd, ‘Linguistics and the Computer Analysis of Tonal Harmony’, *Journal of Music Theory*, Vol. 12, N. 1, Spring, (1968), p. 6.

<sup>8</sup> *Ibid.*, p. 36.

<sup>9</sup> Johan Sundberg and Bjorn Lindblom, *Generative Theories in Language and Music Descriptions*, reproduced from the *Cognition Journal* (1976) in *Machines Models of Music* (Cambridge, MA: The MIT Press, 1993), p. 282.

representations by which the piece is transmitted.’<sup>10</sup> That mentally constructed entity was conceived by Lerdahl and Jackendoff as organized in segmented units of different sizes and types which are organized hierarchically. They proposed to represent hierarchical relationships among pitches by means of what they called *reductions*. This reductive process of pitch analysis consisted in a recursive sequence of analytical simplifications in which, at each stage, the analysis was reduced to a structurally more important or essential material. At the end, the piece could be represented by a single event. They notated this type of analysis by means of a tree-like structure which was written over the score of the music analysed. Here is a simple analysis-tree:



Branches in this tree-like structure functioned as hierarchical subordinates to higher branches. A similar technique, which they called *time-span reduction*, was used to express the way in which pitches are organized in time.

Other models for hierarchical musical analysis have been proposed by several authors, such as James Tenney, who in 1980 proposed in an article entitled *Temporal Gestalt Perception in Music*<sup>11</sup> a theory for analysing scores focusing on the temporal aspects of perception which inspired the creation by Larry Polansky and David Rosenboom of a real-time compositional environment called *Hierarchical Music Specification Language (HMSL)*,<sup>12</sup> or Stephen Smoliar’s<sup>13</sup> experiments in computer analysis based on the analysis model theorized by Heinrich Schenker.

The study of theories and models of analysis is fascinating, and would be worth exhaustively cataloguing. For the purpose of this essay it is enough to notice that these models of musical analysis opened new avenues into the field of computer generated music and provided the framework needed by researchers in order to transfer a more

<sup>10</sup> Fred Lerdahl and Ray Jackendoff, *A Generative Theory of Tonal Music* (Cambridge, MA: The MIT Press, 1983), p. 2.

<sup>11</sup> James Tenney with Larry Polansky, ‘Temporal Gestalt Perception in Music’, *Journal of Music Theory*, Vol. 24, N. 2, Autumn, (1980), pp. 205-241.

<sup>12</sup> David Rosenboom and Larry Polansky, *HMSL (Hierarchical Music Specification Language): a real-time environment for formal, perceptual and compositional experimentation*, Proc. ICMC, San Francisco, CMA (1985).

<sup>13</sup> Stephen Smoliar, ‘A Computer Aid for Schenkerian Analysis’, *Computer Music Journal* 4,2 (Summer 1980), pp. 41-59.

sophisticated or *human-based* inductive process to the computer. The reductive hierarchical process described by Lerdahl and Jackendoff was susceptible of being computerised, and among the experiments that assumed the challenge of codifying the hierarchical models of musical analysis was one carried out by Kemal Ebcioglu at the Department of Computer Science of the State University of New York in 1986. His experiment can be summarised as the mechanization of the Schenkerian analysis with the purpose of transferring to the computer the ability to harmonize chorales in the style of J. S. Bach. The program received as input a cantus firmus and output a harmonization of it along with a hierarchical analysis.

### 2.3 SYNTHESIZING STYLE: DAVID COPE

In 1981 David Cope, professor at the University of California in Santa Cruz, began to work on his *Experiments in Musical Intelligence (EMI)*. It is a project that has been continuously extended and revised over more than a decade and that has ‘delighted, angered, and provoked’<sup>14</sup> the audiences that have listened to its results.

Cope believes that ‘much of what happens in the universe results from recombination’,<sup>15</sup> and that the same can be thought about music, which is conceived by him as the product of several layers of recombination. He placed his experiment into a category of music which he called *Virtual Music*; ‘a broad category of machine-created composition which attempts to replicate style but not the actual notes of existing music’.<sup>16</sup> The program that he wrote receives as input two or more musical pieces of the same or different styles, and generates new pieces with the same style of the given music or, in case that music of different styles were provided, a combination of different styles. He experimented with generating of music in the style of certain composers like Bach, Mozart, Beethoven, and Prokofiev, and also with the combination of different styles, such as Mozart’s piano style combined with Javanese Gamelan music. Some of the pieces generated by his system were published in a set of four CDs called *Virtual Bach*,<sup>17</sup> *Bach by Design*,<sup>18</sup> *Virtual Mozart*,<sup>19</sup> and *Classical Music Composed by Computer*.<sup>20</sup>

---

<sup>14</sup> David Cope with commentary by Douglas Hofstadter and with perspectives and analysis by Eleanor Selfridge-Field, Bernard Greenberg, Steve Larson, Jonathan Berger, and Daniel Dennett, *Virtual Music; Computer Synthesis of Musical Style* (Cambridge, MA: The MIT Press, 2001), p.138.

<sup>15</sup> *Ibid.*, p. 1.

<sup>16</sup> *Ibid.*, p. 3.

<sup>17</sup> David Cope, *Virtual Bach*, Centaur Records, 2003, CRC 2619.

<sup>18</sup> David Cope, *Bach by Design*, Centaur Records, 1993, CRC 2184.

Douglas Hofstadter, professor of cognitive science at the College of Arts and Sciences at Indiana University Bloomington and director of the Centre for Research on Concepts and Cognition of the same institution, wrote several quatrains in reaction to Cope's experimentation in musical style. Here are a few of them:

When music's been treated  
By the likes of Dave Cope,  
Is the mystery banished,  
Or is there still hope?<sup>21</sup>

Or:

One mazurka by EMI<sup>22</sup>  
Has lodged, I confess,  
In the grooves of my brain,  
Causing shame and distress.<sup>23</sup>

And:

Is music, like chess,  
A wee, hard-edged domain,  
Algorithmically handled  
By a pea-sized subbrain?<sup>24</sup>

Lastly:

That's Prokofiev's fate.  
The poor chappie expired,  
Tenth Sonata half done...  
EMI finished it. She's hired!<sup>25</sup>

These rhymes, although they may be considered rather unusual within the context of computer music experimentation, condense some questions that might come up to anybody listening to the music generated by *EMI* knowing that it is automatically generated by a computer.

Another humorous comment on Cope's work was done by Steve Larson, professor of music theory at the University of Oregon, in the form of a series of letters that he, in the fictional role of a counterpoint teacher, sent to a composition a student called

<sup>19</sup> David Cope, *Virtual Mozart*, Centaur Records, 1997, CRC 2452.

<sup>20</sup> David Cope, *Classical Music Composed by Computer*, 1997, CRC 2329.

<sup>21</sup> Douglas Hofstadter, *Staring Emmy Straight in the Eye; And Doing My Best not to Flinch*, included as chapter 2 in David Cope's *Virtual Music* (Cambridge, MA: The MIT Press, 2001), p. 58.

<sup>22</sup> *EMI* is an acronym that stands for Experiment in Musical Intelligence, referring to Cope's system.

<sup>23</sup> Douglas Hofstadter, *op. cit.*, p. 64.

<sup>24</sup> *Ibid.*, p. 66.

<sup>25</sup> *Ibid.*, pp. 74-75.

*Emmy*.<sup>26</sup> In the first letter he congratulated *Emmy* for her fifteen Bach-style inventions in which she surpassed the quality of those composed by other students who in the period of ten weeks have been able to write only one or two. Throughout these letters, however, he tried to make an analysis of those inventions and a comparison with those originally written by Bach in order to suggest some enhancements to the way in which *Emmy* composes.

In order to develop his system of automatic musical composition and style imitation, David Cope incorporated and extended most of the concepts previously described in this paper: in particular the hierarchical analysis models proposed by theorists like Otto Laske, Lerdahl and Jakendoff, James Tenney, or Stephen Smoliar. *SPEAC* is the name that Cope gave to the musical analysis model that integrates his *Experiment in Musical Intelligence* program. According to his words, ‘it provides another level of abstraction for describing the logical motions of musical notes, harmonies, and motives.’<sup>27</sup> *SPEAC* is an acronym in which each of its letters represents an *identifier* used for the musical analysis: *statement, preparation, extension, antecedent, and consequent*. Cope’s system uses those identifiers to hierarchically detect grammatical functions at different musical levels. An exhaustive description of this analytical model along with the explanation of its implementation into his computer program, (including programming code, diagrams, and examples) was written by Cope in his book *Computers and Musical Style*,<sup>28</sup> published in 1991.

Once the syntactical analysis of given pieces of music is done using the *SPEAC* system, Cope’s program scans the inputted pieces in order to find *signatures*. A *signature* is, according to Hofstadter, ‘a characteristic intervallic pattern that recurs throughout a composer’s oeuvre, the use of which lends a high degree of seeming authenticity to a freshly composed piece.’<sup>29</sup> These signatures are stored in a database for later usage in the generative stage of the process.

Once the complete analysis has been done, including the *SPEAC* process and the pattern-matching scanning for *signatures*, all the information derived from the analysis is stored in a database and the re-combinatory process is ready to start. This process *regeneration* of new pieces is described in a chapter called *Style Replication* in his book

---

<sup>26</sup> Steve Larson, *Dear Emmy: A Counterpoint Teacher’s Thoughts on the Experiments in Musical Intelligence Program’s Two-Part Inventions*, included as chapter 13 in David Cope’s *Virtual Music* (Cambridge, MA: The MIT Press, 2001), pp. 237-262. *Emmy*, in Larson’s paper, is a fictional character representing the *Experiments in Musical Intelligence* program (EMI) written by David Cope.

<sup>27</sup> David Cope, *Computers and Musical Style* (Oxford: Oxford University Press, 1991), p. 34.

<sup>28</sup> *Ibid.*, pp. 27-70.

<sup>29</sup> Douglas Hofstadter, *op. cit.*, p. 48.

*Computers and Musical Style*.<sup>30</sup> It basically consists in the generalization of rules inducted from the data recollected throughout the precedent analysis and their application for generating new pieces. Cope's program also provides a visual user interface through which it is possible to control certain parameters of the process. His program provides the ability to automatically replicate and mix musical styles.

Cope's *Experiments in Musical Intelligence* represents one of the most sophisticated instances of experimentation into the subject that was initiated by Brooks et al. in 1957 with the purpose of transferring to the computer not only deductive capabilities, but also and principally, inductive ones.

## 2.4 WHERE IS THE COMPOSER?

These last experiments and Cope's in particular, only make it harder to determine the place of the composer and his, or her, *or its*, decisions. Where is the composer in Cope's experiment? Who is the composer? Can the re-combinatory process be considered a creative process, a process of musical composition?

In the experimental approach described in the first chapter of this paper it was clear that composers had an important role defining the rules that set the framework for the computer to assist them in the compositional process. Xenakis, at bottom, composed his own, human, music using the computer to assist him. But could it be said the same about this last experiment in musical intelligence? Can we see David Cope, in relation to his experiment, as a composer in the same way as we see Xenakis in relation to his own one?

Furthermore, another question arises; what, in essence, is musical composition? Is it, as proposed at the beginning of this paper, just a process of decisions, regardless of *who* takes those decisions? If we assume that the pieces generated by *EMI* are *freed* from human decisions, can we consider them as musical compositions?

The program written by Cope concentrates in it an astonishing sum of knowledge, practices, analysis and theories, which come from many fields and which are exceptionally integrated with a musical purpose. But, still, can we consider *Emmy's* fifteen inventions or her contribution to the completion of Prokofiev's tenth sonata as music?

---

<sup>30</sup> David Cope, *op. cit.*, 1991, pp. 89-140.

Why is it so difficult to find an answer to these questions? I think that that difficulty comes from the meaning that we give to concept of musical composition. If we take for granted that composition is merely a combination of inherited materials then we can logically think that *Emmy* is a composer. Otherwise, if composition involves, also, the *creation* of new ideas and styles, it would be more difficult to consider *Emmy* as a composer. David Cope was conscious about this when he wrote that ‘composers invest in each of their works a combination of new and inherited materials. New materials are created by inspiration. Codifying inspirations in terms of computational means seems improbable, even impossible.’<sup>31</sup>

---

<sup>31</sup> David Cope, *A Computer Model of Music Composition*, published as chapter 18 in *Machine Models of Music*, ed. by S. M. Schwanauer and D. Levitt (Cambridge, MA: The MIT Press, 1993), p. 403.

## 3 ORGANIC COMPOSITION

---

One of the greatest scientific challenges of the twenty-first century is to gain a better understanding of the functioning of the brain when engaging in musical activities.<sup>1</sup>

It seems, according to the researches described in the previous chapter, that the computer could potentially be a composer. But could it be a *creative* composer? Could a computer be able to participate in the conception of new styles in the same way as Beethoven or Schoenberg did? Or is it just limited to a hypothetically unlimited capacity of analysis and recombination?

There was a common element among all the experiments previously described in this paper: the use *rules*. Initial experimentation in the field imposed the rules to the computer while in subsequent experiments the computer was programmed to infer rules by itself through an analytical process. However, before the process of generation of new music started, there was always a framework of rules available for the computer regardless of if it has been obtained by imposition or by analysis.

Does the composer's brain function in the same way? Are we always constrained by rules, imposed or inherited? How did Debussy or Scriabin, among others, change some of the rules?

Somehow, all experiments described in this paper assumed that the computer is an artificial entity and that the only way that it had for imitating human creativeness was through a logical formalization of the creative decisions made by humans. That formalization resulted always, whether imposed or inferred, in a set of rules that the computer used to extract order from an initial chaotic ground. And also, as David Cope wrote, 'even purely machine-composed music is born of programs created by human inspiration'.<sup>2</sup>

What would happen if, instead of formalizing rules by means of programs, a computer is programmed to behave like an autonomous living organism? Would it be possible, in that case, to imagine that such a living organism could evolve and attain creativity?

---

<sup>1</sup> Eduardo Reck Miranda, *Composing Music with Computers* (Oxford: Focal Press, 2001), p. 101.

<sup>2</sup> David Cope, *Computers and Musical Style* (Oxford: Oxford University Press, 1991), p. 34.

Let me add to this essay, prior to an attempting a conclusion, three possible directions that might be taken by current and future researchers and composers.

### 3.1 CELLULAR AUTOMATA

The *Oxford English Dictionary* defines the word *automaton* as ‘something which has the power of spontaneous motion or self movement’. A cellular automaton is a model that produces variations in the states of an array of elements (each element is referred to as a cell), by means of the application of predefined rules which are usually driven by the comparison of the states of each cell with the states of its neighbours cells. All cells in the array are updated according to the same set of rules and each cell may assume two or more states. A very simple example of a one-dimensional cellular automaton with cells that, for the sake of simplicity, can assume only two states (0 and 1) is described next:

The rule to be applied defines that for each step, the state of a cell must be changed only if the state of the cell to the left (in the previous step) was different to 0 or was undefined.

Step 1: 01110101  
Step 2: 11001111  
Step 3: 00101000  
Step 4: 10111100

The states of the cells in the first step are arbitrary defined. The first cell in step 2 was changed from 0 to 1 because the state of the cell to the left in the step 1 was undefined (there was no cell to the left of the first cell). The second cell in step 2 remained in state 1 because the first cell in step 1 was in state 0 and, according to the rule, the state must change only in if the cell to the left is different to 0. The third cell in step 2 was changed from 1 to 0 because in step 1 the second cell was in state 1 and, according to the rule, if the state of the cell to the left is different to 0, the state of the evaluated cell must change. All cells were passed through the same evaluation in order to obtain the resulting configurations of the four steps.

The process of a cellular automaton is iterative, and it can be easily implemented using a computer language. It is interesting to note that, as a result of the process, each

step contains a different configuration of states and that it is not easy to infer their relations without knowing the rules that have been applied throughout the process. Those rules can be very simple, as in the case of the automaton exemplified, or very complex. Complex rules can sometimes produce the effect of an autonomous organism. An example of that illusory effect is shown by a cellular automaton devised by the British mathematician John Horton Conway<sup>3</sup> in 1970. His algorithm, called *The Game of Life*, when graphically represented, creates the visual illusion of two moving objects which, through a sort of reproductive attitudes, produce an offspring of new objects.

Even though the cellular automata model produces rich and interesting structures which may be perfectly applied to musical composition in many different ways,<sup>4</sup> it does not seem to provide the elements for a novel research into artificial musical composition since it is completely dependant upon specified rules, and the autonomy that it seems to obtain is merely an effect or illusion. Also, as the model is applied to the structure of music rather than to the structure of the compositional process, there seems to be no relation between this model and the cognition challenges introduced in the previous chapters. Nevertheless, cellular automata might prove to be a very interesting model when used as a technique for musical composition.

### 3.2 GENETIC ALGORITHMS

*Genetic algorithms* are algorithms that use methods derived from biological processes; in particular those processes related to the evolutionist theory proposed by Charles Darwin in 1859.<sup>5</sup>

Similarly to cellular automata, genetic algorithms modify the configurations of *colonies* of entities throughout the different steps involved in the process; however, genetic algorithms are not based on rules. Once an initial *population* of entities is defined and an objective is established, the genetic algorithm iterates through the processes of *recombination*, *mutation*, and *selection*, until the objective is achieved. The criterion for implementing those three processes can vary according to the application for which the genetic algorithm is intended. In general, the *selection* process implies the choice of certain individual entities from the population in accord to

---

<sup>3</sup> John Horton Conway, *Winning Ways for Your Mathematical Plays*, Volume 2 (Natick, MA: AK Peters, Ltd., 2<sup>nd</sup> edition, January 1, 2003).

<sup>4</sup> See Miranda's *Cellular Automata Musical Engine* described in his book: *op. cit.*, p. 127.

<sup>5</sup> Charles Darwin, *On the Origin of Species by Means of Natural Selection, or the Preservation of Favoured Races in the Struggle for Life* (London: John Murray, 1859).

certain fitness criteria. Once the fittest individuals are selected, they are recombined to produce new individuals which inherit the attributes of their parents, and, in order to allow enhancements in the new generation, certain *mutations* are introduced in the new individuals. In each of the iterations of the process a new generation is created and, by selecting and recombining only the fittest individuals, each new generation should be, on the large scale, more adapted and prepared to accomplish the specified objectives. The logical core of genetic algorithms resides in the criteria and implementation of the processes of *selection*, *recombination*, and *mutation*.

Genetic algorithms, similarly to cellular automata, could be applied to musical composition. It is possible to imagine, for example, a system in which a starting and an ending harmony are defined. The ending harmony would represent the target to be accomplished, and the algorithm, starting from the initial harmony, would iterate through the process of selection, recombination, and mutation, until the final harmony is reached. All intermediate harmonies could be used in the composition as a kind of evolutionary transition between the initial and final harmonies.

In a genetic algorithm process, the human does not define rules; nobody tells the algorithm *how* to obtain the desired results. The only things defined by humans are the targets, and the criteria of *selection*, *mutation*, and *recombination*.

Although genetic algorithms seem to be freed from the rule-based process, they also represent, as is the case of cellular automata, an approach in which the organic behaviour is simulated in the music and not in the compositional process.

### 3.3 NEURAL NETWORKS

A different approach to organic musical composition comes from the research into the possibilities of transferring the *physicality* of the human brain to a computer program.

Such a task would require, firstly, a deep understanding of the way in which our brain functions in physical terms, particularly when engaging in musical activities. Eduardo Reck Miranda, in his book *Composing Music with Computers*, presents a brief introduction to the functioning of the human brain in which he describes the brain as a ‘massive and dense network of interconnected neurones’.<sup>6</sup> He also points out that the brain can be studied as ‘a massive array of parallel processing devices: the neurones’.<sup>7</sup> That last concept insinuates the idea of a possible codification of neural networks into

---

<sup>6</sup> Eduardo Reck Miranda, *op. cit.*, p.104.

<sup>7</sup> *Ibid.*, p. 106.

computer programs and, ultimately, the possibility of replicating certain aspects of the composer's brain in a computer.

Imagine that it would be possible to completely replicate the mechanics of the human brain through a computer program. Would such a program be able to compose music in the same way as humans do? I do not have an answer for that question; however, I have one last doubt: do we compose music only by means of our brains?

## 4 F I N A L T H O U G H T S

---

Just as the wheel was once one of the greatest products of human intelligence, a mechanism which allowed one to travel farther and faster with more luggage, so is the computer, which today allows the transformations of man's ideas.<sup>1</sup>

Fifty years of research into computer-assisted musical composition are enough to show that the interaction between humans and computers stimulates the transformation of musical ideas and the development of a series of skills which I consider will be fundamental to the composer of this century.

Composers willing to use computers for sophisticated compositional purposes will need to get involved with the technology available for that purpose. They will need, also, to learn how to write computer programs. Today, technology is much friendlier and accessible than fifty years ago and there is a strong tendency towards the use of high-level<sup>2</sup> programming languages which are much easier to learn than low-level ones. These high-level programming languages usually use the object-oriented paradigm,<sup>3</sup> which supports reusability, inheritance, and modularization of objects. Five decades ago, each experiment implied the development of all its components from scratch; today, by reusing and extending objects constructed by others, composers and researchers can focus on their specific objectives without needing to build everything from scratch. Nevertheless, technology stills requires the development of logical and mathematical skills which might prove to be useful not only for programming computers but also for understanding and creating music.

The use of computers for musical composition stimulates also the development of analytical skills. Excluding the latest researches into genetic algorithms and neural networks, it can be said that the computer needs to have a framework of logic to be able to assist the human in the compositional process. The construction of those frameworks requires a deep analysis and formalization of musical styles and systems. A composer working with a computer will need to develop not only the ability to realize profound

---

<sup>1</sup> Iannis Xenakis, *Formalized Music; Thought and Mathematics in Composition* (Bloomington: Indiana University Press, 1971), p. 132.

<sup>2</sup> A high-level programming language provides a higher level of abstraction from the basic machine language than low-level programming languages. High-level programming languages tend to represent machine operations through symbols and structures understandable by human beings.

<sup>3</sup> A clear introduction to the object-oriented paradigm is published by Sun Microsystems at <http://java.sun.com/docs/books/tutorial/java/concepts/index.html>.

musical analysis but, also, the capacity of formalizing that analysis in order to apply it to an artificial generative process.

Another important skill stimulated by the use of computers in the compositional process is the capacity of self-knowledge. The only archetype of composer available to be emulated by a computer program is the human one, and in the attempt to emulate compositional abilities in a computer the composer will need to understand fully how it is that humans compose and perceive music.

The use of computers for musical composition generates, also, a series of philosophical issues related to the nature of musical composition; similar to the ones that will remain unclosed in my essay. Every composer working with computers will be exposed to those issues in some way or another.

## **4.1 BEYOND DECISIONS**

The relation between the composer and the musical decisions has been significantly modified by the use of computers in the compositional process. When composing with computers, the decisions traditionally taken by composers such as choosing pitches, rhythms, durations, or dynamics, are not more involved in the creative process, and the composer focus entirely on the formal structures of music. In that sense, the composer's role, when working with computers, stands beyond the decisions. His new role consists in the creation of decision-making frameworks for the computer, and the creation of those universes of rules, conditions, and procedures, can be considered as a creative and artistic process in itself.

As I mentioned at the beginning of this essay, the new role for the composer consists in the composition of compositional systems.

## **4.2 COMPUTER-ASSISTED MUSICAL COMPOSITION**

Through the analysis of the experiments described in this essay, it is possible to outline the following historical direction in relation to the types of compositional behaviours programmed in computers:

*dependent → analytical → autonomous*

The title of this essay uses the words *assisted composition* as a way of defining a type of relation between the composer and the computer within the compositional process. If we remove the word *assisted*, that relation ceases to exist.

The challenge for the future research into computer composition will consist in obtaining a balanced relation between autonomy and assistance. As long as the computer's autonomy implies a richer capacity of assistance, the relation between composers and computers will continue to grow and the possibilities and resources for extending the formal structures of music will be enlarged.

# I N D E X

---

- |                                      |                        |                                       |                |
|--------------------------------------|------------------------|---------------------------------------|----------------|
| Bach, J. S. ....                     | 19, 21                 | MIT .....                             | 17             |
| basso continuo .....                 | 9                      | Mozart, W. A. ....                    | 19, 20         |
| BBC .....                            | 7                      | mutation .....                        | 26, 27         |
| Beethoven, L. v.....                 | 19, 24                 | Neumann, P. G.....                    | 13, 15         |
| Brooks, F. P. ....                   | 13, 15, 16, 22         | neural networks.....                  | 27, 29         |
| cellular automata.....               | 25, 26, 27             | object-oriented paradigm .....        | 29             |
| Chomsky, Noam .....                  | 17                     | Pegasus Computer .....                | 7              |
| Conway, John H. ....                 | 26                     | Polansky, Larry.....                  | 18             |
| Cope, David.....                     | 19, 20, 21, 22, 23, 24 | Prokofiev, S. ....                    | 19, 20, 22     |
| Debussy, C.....                      | 24                     | Reck Miranda, E. ....                 | 24, 26, 27     |
| Ebcioglu, Kemal .....                | 19                     | recombination .....                   | 19, 24, 26, 27 |
| Fux, J. J.....                       | 7                      | Romanticism .....                     | 6              |
| Gamelan.....                         | 19                     | Rosenboom, David .....                | 18             |
| genetic algorithms .....             | 26, 27, 29             | Rothgeb, John .....                   | 3, 9, 10       |
| Gill, Stanley.....                   | 7, 8, 9                | Saint-Lambert, M. de.....             | 9, 10          |
| grammars .....                       | 16, 17                 | Schenker, Heinrich .....              | 18             |
| Greeks.....                          | 7                      | Schoenberg, Arnold .....              | 8, 24          |
| Harvard .....                        | 7, 13                  | Scriabin, A. ....                     | 24             |
| Heinichen, J. D. ....                | 9, 10                  | serial composition.....               | 8              |
| hierarchical analysis .....          | 17, 18, 19, 21         | signatures .....                      | 21             |
| Hiller, Lejaren A.....               | 2, 5, 6, 7, 8, 11      | skills.....                           | 10, 29         |
| HMSL.....                            | 18                     | Smoliar, Stephen.....                 | 18, 21         |
| Hofstadter, Douglas .....            | 19, 20, 21             | State University of New York .....    | 19             |
| Hopkins, A. L. ....                  | 13, 15                 | Stravinsky, Igor.....                 | 7              |
| Illiack Computer .....               | 5, 7                   | Sundberg, Johan.....                  | 17             |
| Indiana University Bloomington ..... | 20                     | Tenney, James.....                    | 18, 21         |
| Isaacson, Leonard M.....             | 2, 5, 6, 7, 8, 11      | University of Calif. Santa Cruz ..... | 19             |
| Jackendoff, Ray .....                | 17, 18                 | University of Illinois.....           | 5, 7           |
| Larson, Steve .....                  | 19, 20, 21             | University of London.....             | 17             |
| Laske, Otto E. ....                  | 16, 17, 21             | Virtual-music .....                   | 19, 20, 21     |
| Lerdahl, Fred .....                  | 17, 18, 19, 21         | Winograd, Terry .....                 | 17             |
| Lindblom, Bjorn .....                | 17                     | Wright, W. V. ....                    | 13, 15         |
| Markov chains .....                  | 7, 14                  | Xenakis, Iannis .....                 | 10, 11, 22, 29 |
| Markov, Andrei A. ....               | 14                     |                                       |                |