Some Projects and Reflections on Algorithmic Music

Rubén Hinojosa Chapel

Music Technology Group, Universitat Pompeu Fabra
Ocata 1, 08003 Barcelona, Spain
+34 93 542 2104
ruben.hinojosa@iua.upf.es

Abstract. Algorithmic Music Composition with computers (in real and non-real time) has had many approaches. It can be viewed from different points of view: scientific, technological, artistic or philosophical. In this paper the author introduces three projects developed between 1990 and 2000 in Havana, during his time as a professor, researcher and composer at the University of Arts of Cuba. He also exposes some philosophical reflections on algorithmic music systems, derived from his work at Havana and his current research in the Music Technology Group of the Universitat Pompeu Fabra.

Keywords: Algorithmic Composition, Real-Time Composition, Interactive Music Systems, Virtual Instruments, Computer Aided Composition, Algorithmic Music.

1 Introduction

In 1989 Cuban composer Carlos Fariñas (1934-2002) founded, with some colleagues, the *Estudio de Música Electroacústica por Computadora* (Studio for Electroacoustic and Computer Music) of the *Facultad de Música*, at the *Instituto Superior de Arte* (University of Arts) in Havana. One year later I started collaborating with the Studio as an undergraduate student of Computer Science.

From the beginning I was assigned a project on Algorithmic Music. Our knowledge on the field was almost null, so we started from scratch, provided only with a couple of articles, some historical and anecdotal references (Hiller, Xenakis...) and a great enthusiasm in doing our best. For one decade I have been working on Algorithmic Music composition and software development and experimentation. Now I have some experiences and reflections that would like to share with other people interested in the field.

2 Musical Fractals (1990-1994)

Our first idea was to create a system for generating musical structures automatically, a system without a direct connection with any known musical style. Nevertheless, as we will see later, we implemented melodic transformations from traditional counterpoint, and even new ones.

Fractal images were very popular at that moment; the musical experience based on fractals carried out by American composer Charles Dodge [1] was a starting point for

our research. Dodge suggested a musical structure based on a metaphoric interpretation of the self-similarity concept. He departed from the Koch curve for building parallel voices that contain proportional relationships between them, similar to those existing between the triangles of the Koch curve.

We elaborated and implemented an algorithm based on the Dodge's interpretation of self-similarity. Our first system, *Musical Fractals*, needs as a seed data, a melody, a list of melody transformations, and some numerical values such as: number of voices, values that will affect the relationships between them, etc. It computes the "piece" in non-real time, generating up to four parallel voices. One of the voices is the original melody and its variations. We implemented some interesting features that proved, through practical experiences, its strength and weaknesses. Some of these features are [2]:

1. **Scales.** The program is able to use up to fifteen different musical scales, even a user defined one, for computing the whole "piece".
2. **Traditional melodic variations.** Melodic transformations from classical counterpoint are algorithmic procedures used along centuries of musical tradition. They are powerful tools for developing a melody, so we decided to test their potential inside a computer program.
3. **Non-traditional melodic variations.** Some non-traditional melodic variations were implemented, following very personal approaches. They are:
   – **Addition:** Randomly adds some notes to the melody without affecting the total length.
   – **Subtraction:** Randomly deletes some notes from the melody without affecting the total length.
   – **Reverse time:** It is like traditional "reverse", but it only reverses the note durations of the melody.
   – **Reverse pitch:** It is like traditional "reverse", but it only reverses the note pitches of the melody.
   – **Generation:** Uses a 1/f fractal noise generator for changing each pitch of the melody.
   – **Simulation:** Uses a particular approach, based on *Markov Chains*, for changing each pitch of the melody. The resulting melody sounds a little bit like the original one.
   – **Arpeggio:** Changes the notes whose duration is greater than or equal to a quarter-note, by an arpeggio of four notes, without affecting the total length of the melody. The algorithm uses interval values provided by the user.
   – **Logarithmic:** Changes every pitch by a new one, computed with a personal algorithm that uses the logarithmic function, and involves some existing pitches.

An interesting feature that proved good results is the possibility of applying not only isolated melodic variations to the melody, but a set of joined variations that conform a much complex transformation. For instance, think about applying the following variations in order: augmentation, arpeggio, simulation and diminution. The resulting melody is only one, not four. Of course, it is possible to obtain four different melodies too! A simple command interpreter was implemented for entering coding of complex transformations. Composers found this possibility as a new and powerful compositional tool [2].
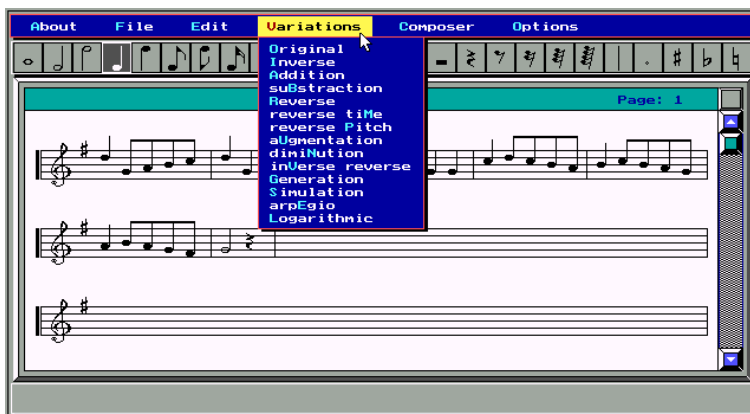
Fig. 1. A screenshot of Musical Fractals

A particular effort was the creation of an embedded score editor (by Claudio Daniel Ash) for entering the melody and for another upcoming projects. The resulting "piece", as well as the original melody and every transformation can be heard through a Roland MPU401 MIDI interface and an external sound module. Additionally, they can be saved to a Jim Miller's *Personal Composer* MIDI file.

Musical Fractals runs under MS-DOS and never was ported to MS Windows. It was awarded with some prizes. This project provided us with our first experiences in experimenting computer algorithms for music composition. One electroacoustic music piece was composed by Carlos Fariñas *(Cuarzo: Variaciones Fractales)*, which was premiered in 1991 in Havana, in the frame of the Festival of Contemporary Music [2], and has been played abroad. The author made also, in 1994, a short electronic piece entitled *ET llamando a casa*, which demonstrates some of the features of the computer music system.

2.1 Orbis Musicae (1993-1996)

In 1993 we started another project with new goals in mind. We were faced with the problem of creating an interactive music system for real-time performances. The idea came about through several ways, but a very influencing one was our personal meeting with Dr. Max Mathews in 1991, during the International Festival of Electroacoustic Music held in Varadero beach. There I had the opportunity to talk with him. Among several questions, I posed this one: How would you use a music made algorithmically? Mr. Mathews kindly answered:

"I would be interested in keeping an interaction with the algorithm; a part from the computer and a part from myself. I am interested in algorithms for improvising. With these algorithms, the musician and the computer play the music together. The algorithm chooses the notes, but the musician can select, among the options given by the program, the one he likes [3]."

With these ideas in mind we gave birth to *Orbis Musicae*, which acts like an instrument, where the musician controls different and variable parameters in real-time while the music is computed. The algorithm is quite straightforward:

Fig. 2. A screenshot of Orbis Musicae

There are twelve planets around the Sun moving each one on an elliptic trajectory. At the beginning each planet is assigned a grade from the chromatic scale. Then one or more triangles are placed over the orbital plane. When the planetary system starts moving, one or more planets visit the area of the triangles. As soon as a planet goes in or out from a triangle, a Note On or Note Off message is triggered, sounding on or off a MIDI controlled external sound source. The next time this planet goes inside the triangle, the original pitch assigned to it could remain the same, or could be changed to a new one, according to the initial choice of the musician. Each triangle has assigned a particular MIDI channel, so different MIDI programs (instruments) or sound modules can be controlled at the same time [4].

The musician can change the position and speed of the planets during the real-time performance. The configuration of the system, say: planet positions, planet speeds, assigned pitches, triangles and its assigned MIDI channels, can be saved internally at any time, and can be restored also whenever the user wants. *Orbis Musicae* uses ten memory banks, and follows the "total recall digital mixers" philosophy. The composer can use this feature for creating a scheme of configurations, which would be useful for planning the development of his piece in a sort of a score.

Orbis Musicae runs under MS-DOS and, as well as its predecessor, never was ported to MS Windows. An electroacoustic music piece was composed by Carlos Fariñas, who used its real-time capabilities for recording fragments of music in a sequencer. Later he took these fragments for creating a tape composition *(Orbitas Elípticas)*. This music work was played in 1994 in the Bourges's International Festival of Electroacoustic Music. Another electroacoustic music work was created by Cuban composer Roberto Valera, who used the software for his real-time piece *Hic et Nunc*, performed for the first time in 1996, with the assistance of myself, in the frame of the Havana's Festival of Contemporary Music.

Orbis Musicae has two essential properties:

1. It is a self-regulated system that has a personal behaviour. It can play itself endlessly without any human intervention.

2. The task of the human player is to influence the behaviour of the system, as if he were an instrument player. In fact, he is an instrument player. A player of a new kind of instrument, an active instrument. Traditional instruments always play a passive role, they react to the human gesture, but they are unable to offer the musician any musical idea by itself. At that time we used to name this kind of software active instrument, "virtual instrument".

In the middle of our investigation, we found previous experiences from other researchers whose works connect deeply to, and reinforce, the ideas we were working on. These experiences come, in one hand, from Louis and Bebe Barron, and on the other hand, from John Bischoff and Tim Perkis.

2.2 Louis and Bebe Barron

We found very interesting and pioneering the works done by Louis and Bebe Barron in New York, during the fifties of the past century. They intended to build new sonic models using the spontaneous electric evolution of some electronic circuits coupled between themselves, whose oscillation frequencies were placed in the audible range.

The main idea was to build series of active circuits with specific frequencies and transitory regime. By coupling these circuits to each other and influencing the behaviour of its neighbour circuits, it is possible to make changes to its own parameters. According to a partially predictable process, the union of synchronizing influences coming from its neighbour oscillators will modify the state of the oscillations of each circuit, so that they modulate their oscillations between themselves [5].

The first circuit state is dictated by external conditions, which can be changed at will. Leaving it to itself, the system of circuits follows an evolutionary process, which can be defined as the behaviour in reaction to external stimuli. This acoustic behaviour is modified according to the relationship and order established between the circuits, and confers personal characteristics to a particular considered system [5].

If we choose and study conveniently the parameters of those circuits, it could be possible to obtain an interesting sonic result, which could lead to the creation of an electronic music composition. Under this perspective [5], Louis and Bebe Barron made music for the cinema, especially for the films *Bells for the Atlantis, Electronic Jazz* and the science fiction film *Forbidden Planet* (1956). The soundtrack of this film is a wonderful example of artistic and avant-garde creation, and a remarkable example of the musical use of sound synthesis by modulation.

In the works by Louis and Bebe Barron we have found the notions of a self-regulated sound generation system, which owns a personal and autonomous sonic behaviour that can be controlled and changed by external influences in an interactive way. We have found also the same principle in the works by John Bischoff and Tim Perkis.

2.3 John Bischoff and Tim Perkis

On the CD *Artificial Horizon*, recorded between 1989 and 1990 by American composers John Bischoff and Tim Perkis, is exposed a sample of what they call "Music for New Software Instruments". In the CD booklet they express the philosophy of their music in the following terms:

"For us, composing a piece of music is building a new instrument, an instrument whose behavior makes up the performance. We act at once as performer, composer and instrument builder, in some ways working more like sculptors than traditional musicians. (...) There is another feature of the computer that attracts us: its ability to build systems of interaction with complex dynamics, systems only partially predictable, which can develop a unique "body" of their own. These woolly computer instruments can also be designed to respond to players' actions in new ways, creating a music which contains the trace of human gesture, in addition to having a degree of autonomy. In fact, for us, the distinction between composing a new piece of music and building a new instrument is not clear-cut: composing a piece of music for us IS building a new instrument, an instrument whose behavior makes up the performance. We act at once as performer, composer and instrument builder, in some ways working more like sculptors than traditional musicians. And in each case, the focus is on creating a system as open and alive as possible, bearing the precious marks of an individual character." [6]

And specifically talking about his 1978-80 piece *Audio Wave*, John Bischoff says: "AUDIO WAVE was written for pianist Rae Imamura (...). My idea was to make a live computer piece for Rae where both of her hands would be continually active, as in her conventional keyboard playing, but where her actions would serve to influence an ongoing musical output rather than have the task of initiating each sound." [7]

When we knew about the principles behind the works by the Barrons, Bischoff and Perkis, and after looking back to our experiences, we felt that we had found what path to travel through. So, we decided to build an interactive algorithmic music system for real-time performances, not-based on any known musical style, which could act as an active instrument (self-regulated system) "where the user's actions would serve to influence an ongoing musical output rather than have the task of initiating each sound". We were looking for a more flexible Virtual Active Instrument. Then the *Fractal Composer* project was started at the EMEC/ISA.

3   Fractal Composer (1996-2000)

Fractal Composer [8] is intended to be a virtual musical instrument for real-time performances. It plots chaotic attractors, dynamic systems and some related formulas, and makes music from these calculations while the musician introduces changes to musical parameters and listens to the results, all of this in real-time.

The system, which runs under MS Windows, features seven different fractal formulas and related algorithms for tone generation, which combine six ways of mapping pixel colors into pitches, and four note-duration or rhythms. Up to four interdependent voices may be used, conducted through three different manners or styles. Each voice owns its loudness or dynamics, its pitch limits (range) and scale. This program offers twenty-four different scales, including nine user defined ones.

The user may have control over some MIDI functions like: program changes, modulation and panning. Each voice can be moved from left to right or vice versa, automatically, at the speed the user chooses. Although program names are showed in the General MIDI convention, of course it is possible to use any non-GM external sound module. In addition to this, it is possible to load a digital sound file and play it together with the MIDI fractal music.
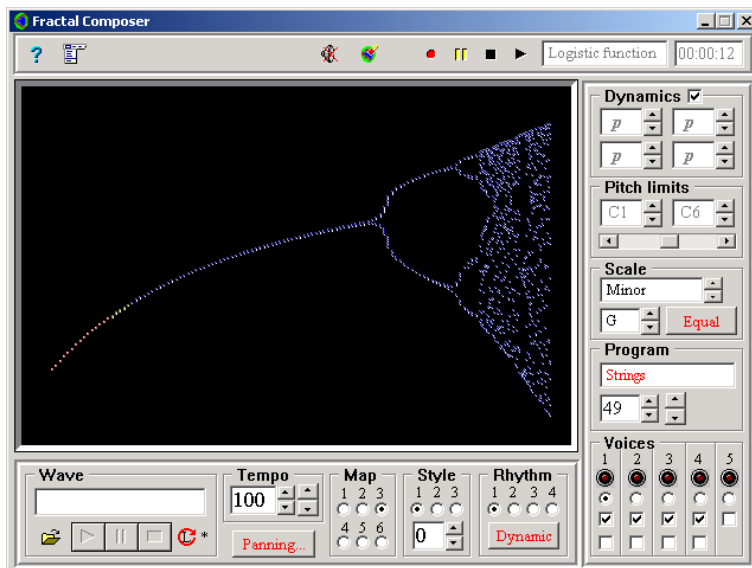
Fig. 3. A screenshot of Fractal Composer

While *Fractal Composer* creates music in real-time, it is possible to save the resulting music, with all the changes (performance) the user has done, in a Standard MIDI File. This lets him edit his music in any sequencer or music notation software that support SMF.

The musician can store all the settings in a configuration file to be recalled later, in another session. This means that the player doesn't lose his fractal type nor its parameters, voices selected, patches, dynamics, scales, and even his own scales. A chronometer appears in the upper right-hand corner of the display to inform the performer about the duration of his piece as time goes by.

As Xenakis said in 1971 in his book *Formalized Music*: "With the aid of electronic computers, the composer becomes a sort of pilot: pressing buttons, introducing coordinates, and supervising the controls of a cosmic vessel sailing in the space of sound, across sonic constellations and galaxies that could formerly be glimpsed only in a distant dream." [9]

The author wrote three electroacoustic music pieces with this system:

1. *El fin del caos llega quietamente*, which is intended to demonstrate that mathematics can also be a path to music. It was created entirely in real-time from the calculus of the *Logistic Function*, and recorded in one pass with no overdubbing.
2. *Satélites*. Basic sonic material was created in real-time from the calculus of the Henon Map. It was premiered in the XII Havana's Festival of Contemporary Music, in 1997.
3. *Oro-iña*, a real-time performance for computer, Afrocuban drum set and two dancers. It was played for the first time in 1998, in the frame of the International Festival of Electroacoustic Music held in Havana.

Fig. 4. Picture of the Oro-Iña performance (Photo: Archie)

4   Some Theoretical Reflections

4.1   In Search of a Satisfactory Algorithm

Algorithmic Composition researchers have tried different approaches for handling the music elements and for generating musical structures. Traditionally one of the most important elements of music has been melody. Many algorithms and models for "composing" melodies have been developed, from 1/f fractal noise to rule based or constraint programming.

Cuban composer Carlos Fariñas (1934-2002) used to say that every melody but those from monodic systems, always has an implicitly harmonic context. So, according to this idea, every random procedure for creating melodies should take this principle into account. It has no sense to look for an algorithm for creating "beautiful" or "inspired" melodies without influencing the random process by a harmonic progression.

On the other hand, when researchers intend to mimic a known music style, it is often known what elements, characteristics or procedures they should model, but what path should be follow in order to generate satisfactory musical structures not-based on any known musical style? Musicology and music composition tradition have the answer. When we were looking for a solution, composer Fernando Rodríguez (Archie) came up to us and replied: "you should try to model Analogy and Contrast".

"The notions of foreground and background (...) are critical in controlling musical flow. If similarity is in the foreground, the listener will perceive the music as continuing uninterrupted; if difference is more prominent, then the perception will be one of contrast. (...) When contrast is in the foreground, it is introduced to avoid boredom, and to deepen the listener's experience. Contrast creates emotional breadth, setting off ideas and heightening relief and definition of character. (...) Musically, when we hear familiar material in new contexts, its meaning is enriched." [10]

These reflections around melody and structure only refer to our western music tradition. It could be possible that they do not match with music traditions from other different cultures.

4.2   In Search of a Definitive Composition System

Throughout the music history, composers have developed techniques, most of them algorithmic procedures, to handle all the elements of music, say: melody, harmony, rhythm, timbre, articulation, form... What a composition system does is to apply these techniques, and even new or personal ones, to the material provided by the user, i.e., the composer. The ways to handle the elements of music are so many, almost infinite so, from our point of view, it is impossible to find a definitive computer composition system [11].

Every music algorithm leaves its fingerprinting in the sonic result of its execution. It has no sense to look for an universal algorithm for composing any known music style. Composers use many algorithms or algorithmic procedures everyday, and the doors for creating new compositional procedures and new music styles are always opened, though it's no easy to travel it through. Music composition involves creativity, which is impossible to lock in a scientific model. It always flies away beyond our imagination.

4.3   Two Reflections About Authorship

In [12] I found an interesting question that made me think: "(...) if an algorithm faithfully represents an artist's creative process, what is the difference between music produced by the artist and music produced by the algorithm?"

Algorithmic composition leads to the following situation: the user gives instructions to a computer to conceive an object (music). After a while, the man receives this object from the machine. So, what now? He says: "this is my own work". Has the man stolen the object from the machine? Does this object belong to the computer?

Do not forget who has mentally conceived that object before its physical existence. Man has thought about that object, with more or less precision, before giving instructions to the machine. So, the computer has the task to give birth to the object dreamed by the man. When an artist designs a monumental sculpture, it is built by several (or even many) workers, but nobody has doubts about the authorship of the sculpture. Who is the author of the *Sagrada Familia* temple? Who denies it is Antonio Gaudí? Who denies the authorship of the *Tour Eiffel* to Gustave Eiffel?

Computers only simulate, through very strict instructions from the man, some elements of the human thinking. During the creative process, they can contribute some things to the task commanded by the man, but they only can contribute things that were thought before, things that were mentally conceived previously by the man. They cannot contribute things unconceived by the man, because they have no will nor awareness. That is the difference between music produced by the artist and music produced by the algorithm.

The man conceives and programs creation strategies, which imitate his possibilities, skills and knowledge. So, his personality will be present in the machine's results. Computers have no special artistic skills or virtuosity. They only have a representation of the skills and knowledge from the man. They are only able to mimic those human properties.

Can a machine express its individuality, its own personality, its own subjectivity? These qualities are not properties of a computer, so they cannot be expressed. Only the man can express his individuality, personality and subjectivity, from the moment he selects and gives instruction to the machine, from the very instant he conceives a music program, or when he configures the options of the software. Machines impregnate with some logic and formal characteristics the result of its computations, but the man is who gives imagination to those calculi, the man is who transmits his human sensibility with the help of a computer, and he is who transforms in art the science that could exists in automatic creations.

I have found also in [12] the following interesting statement: "(...) music produced by algorithmic composition is considered somehow inferior not because it was produced by an algorithm, but because it is someone else's music–it belongs to the *designer* of the algorithm, and not to the user of the algorithm."

If we accept this statement as a valid one, maybe it should be said: *Wozzek* does not belong to Alban Berg (the *user* of the twelve-tone algorithmic procedures) but to Arnold Schoenberg (the *designer*). Traditional non-computer algorithmic methods are really compositional procedures, which are always adapted by the composer to his mental scheme, to his personal point of view about music, and to his own experience and skills.

When the user configures the options of any algorithmic composition system, and gives it the seed data, he transmits his own personality, as well as when he uses any conventional algorithmic procedure, or even a rule-based music composition formalism like traditional counterpoint. So, we firmly believe that music composed with the aid of an algorithmic composition system, belongs to the user.

4.4   Algorithmic Music Composition: Why?

Due to the wide range of possibilities offered by computers and other electronic music devices, which are sometimes exaggerated, it is often though erroneously that usual music knowledge is unnecessary for making music with those equipments. We think computers are a powerful tool for the musician. They will help the artist in developing his ideas, in stimulating his imagination, in speeding up some technical procedures of music composition. Computers enrich the compositional process, but they will not provide the user an unexisting talent. Nevertheless, they are able to stimulate the development of an undiscovered talent or innate musical capabilities [13].

Finally, I would like to point out some general ideas related with algorithmic music composition systems I have compiled:

1. These systems stimulate the composer's creative imagination in a very new and promising way, with lots of possibilities.
2. Composition programs can handle much more data and much faster than a human composer. They let him think in a high level of abstraction, leaving low-level details to the computer.
3. They are a door for searching new aesthetic concepts, new sonic conceptions and new ways of organizing sounds. So, they are a path for music development.
4. These systems allow scientific verification of music theories, when it is intended to simulate a known musical style in order to analyse and study it.
5. They allow to better know how musical processes take place in the human mind, so they let us know better the nature of the human being.

5   Present and Future Work

In 2001 I was granted a scholarship from the *Universitat Pompeu Fabra* in Barcelona, for making my doctor degree in Computer Science and Digital Communication. Now I have a good opportunity to learn new things, to work on new projects and to develop the ideas I have been working on in Cuba since 1990.

In December 2002 was created inside the MTG (*Music Technology Group*) led by Dr. Xavier Serra, the IST (*Interactive Systems Team*). Having Sergi Jordà as Project Manager, the IST is integrated by: Alvaro Barbosa, Gunter Geiger, Martin Kaltenbrunner, José Lozano and myself. Now we work on a new project named *reacTable∗*, which puts together most of the research interests and know-how of the IST members.

ReacTable∗ is a project that activates important interdisciplinary research in the field of Computer Music, which significantly departs from the MTG traditional work based on signal processing techniques. Some of the areas of research involved are algorithmic composition and real-time music creation / composition. For the near future we hope to integrate our experiences in the development of this new project.

6   Conclusions

Algorithmic music composition with computers (in real and non-real time) has had many approaches [9], [14–24]. It can be viewed from several points of view: scientific, technological, artistic or philosophical. We have introduced three projects developed between 1990 and 2000 at the EMEC/ISA in Havana. The first one generates musical structures in non-real time, while the other ones also generate musical structures but in real-time, in an interactive way. Neither is based on known musical styles, though they use basic musical concepts or technical procedures. To my mind, a concluding idea is the development of an interactive algorithmic music system for real-time performances, not-inspired (no mimic) on any known musical style, which could act as an active (self-regulated) instrument "where the user's actions would serve to influence an ongoing musical output rather than have the task of initiating each sound". Finally, we have exposed some theoretical reflections. I hope this paper be useful for the development of discussions and ideas related with the topics discussed here.

Acknowledgments

References

1. Dodge, Ch., Bahn, C. R.: Musical Fractals: Mathematical Formulas Can Produce Musical as well as Graphic Fractals. Byte, June 1986, p. 185-196
2. Hinojosa Chapel, Rubén: Sistemas de Ayuda a la Composición Musical. Thesis work, Faculty of Mathematics and Computer Sciences, University of Havana (1993)
3. Hinojosa Chapel, Rubén: Acerca de los Instrumentos Electrónicos, la Música Electroacústica y las Computadoras. Banco de Ideas Z / Instituto Superior de Arte, Havana (1995)
4. Hinojosa Chapel, Rubén: Sistemas de Ayuda a la Composición Musical: la Experiencia del EMEC. Proceedings of the I International Congress of Informatics on Culture, International Convention INFORMATICA '94, Havana, February 1994
5. Moles, Abraham: Les Musiques Expérimentales. Editions du Cercle d'Art Contemporain (1960)
6. Bischoff, J., Perkis, T.: Artificial Horizon. CD booklet, Artifact Recordings (1990)
7. Bischoff, John: Software As Sculpture: Creating Music From the Ground Up. Leonardo Music Journal, Vol. 1 No. 1, 1991
8. Hinojosa Chapel, Rubén: Fractal Composer: un Instrumento del Siglo XXI. Proceedings of the IV International Congress of Informatics on Culture, International Convention INFORMATICA 2000, Havana, May 2000
9. Xenakis, Iannis: Formalized Music: Thought and Mathematics in Composition. Rev. ed. Stuyvesant, New York, Pendragon Press (1992)
10. Belkin, Alan: A Practical Guide to Musical Composition `http://www.musique.umontreal.ca/personnel/Belkin/bk/4a.html#2)%20Contrast`
11. Hinojosa Chapel, Rubén: Music Generation Panel (A critical review). Workshop on Current Research Directions in Computer Music, Barcelona, Nov 15-16-17, 2001, Institut Universitari de l'Audiovisual, Universitat Pompeu Fabra `http://www.iua.upf.es/mtg/mosart/panels/music-generation.pdf`
12. Jacob, Bruce L.: Algorithmic Composition as a Model of Creativity. Organised Sound, volume 1, number 3, December 1996 `http://www.ee.umd.edu/~blj/algorithmic\_composition/algorithmicmodel.html`
13. Hinojosa Chapel, Rubén: Entre Corcheas y Electrones. Por Esto!, Mérida, Yucatán, Mexico, July 31, 1996
14. Alpern, Adam: Techniques for Algorithmic Composition of Music `http://hamp.hampshire.edu/~adaF92/algocomp/algocomp95.html`
15. Brün, Herbert: From Musical Ideas to Computers and Back. In: Harry B. Lincoln (Ed.), The Computer and Music Ithaca, Cornell University Press (1970)
16. Cope, David: Computers and Musical Style. Madison, WI: A-R Editions (1991)
17. Hiller, L., L. Isaacson: Experimental Music. McGraw-Hill Book Company, Inc., New York (1959)
18. Koenig, G. M.: Project One. Electronic Music Report 2. Uthrecht: Institute of Sonology, 1970. Reprinted 1977, Amstermdam: Swets and Zeitlinger
19. Kunze, Tobias: Algorithmic Composition Bibliography `http://ccrma-www.stanford.edu/~tkunze/res/algobib.html`
20. Maurer IV, John A.: A Brief History of Algorithmic Composition `http://ccrma-www.stanford.edu/~blackrse/algorithm.html`
21. Papadopoulo, G., Wiggins, G.: AI Methods for Algorithmic Composition: A Survey, a Critical View and Future Prospects. `http://www.soi.city.ac.uk/~geraint/papers/AISB99b.pdf`
22. Roads, Curtis: The Computer Music Tutorial. The MIT Press, Second printing (1996)
23. Zicarelli, David.: M and Jam Factory. Computer Music Journal 11(4), 1987, p. 13
24. More references: `http://www.flexatone.com/algoNet/index.html`