Emulating the COMDASUAR

Diego de la Fuente Curaqueo
diego.delafuente@uchile.cl

ABSTRACT

The advent of digital technologies has reshaped the way we conceive musical creativity within last decades, influencing both technical and compositional affairs.

In this field, the "Asuar digital-analog computer" -also known as COMDASUAR- has become a true legend among music composers and electroacoustic researchers from Chile and the world.

The mystery surrounding this particular invention is based on the unfortunate fact that there is no way to access the original source code, that offered several tools designed to generate, process and play musical material.

This paper introduces the historical context in which this piece of technology was conceived. It will also address some original sources that will give us hints about the functionalities and features that the machine included. Finally, we will introduce a prototype of software that emulates these features based on the mentioned sources.

1. INTRODUCTION

In 1978 the engineer and renowned Chilean composer Jose Vicente Asuar conceived an ambitious project in which his two areas of expertise converged: the Comdasuar, the first computer designed in Chile to reproduce, edit and compose music. The original (and unique) prototype of this machine had a short operating life and has not been restored this far. Hence, historical records such as the music album *Así habló el computador* (Also sprach the computer) from 1978; and the article *Un sistema para hacer música con un microcomputador* (A system to create music with a microcomputer) from 1980, are both the main -and only- sources available to understand the complexity of this unusual tool.

Both sources contain examples and descriptions that the Comdasuar algorithm offered to the composer/user in order to create, edit and/or generate musical material. Therefore, the main objective of the project introduced in this paper is to rewrite the Comdasuar algorithm from scratch, based on the mentioned sources, as well as to publish a beta version of a free application that emulates the characteristics of the original program.

2. HISTORICAL BACKGROUND

The Comdasuar, designed by Jose Vicente Asuar, was undoubtedly a pioneering project regarding digital sound and music technology; even more so considering that its development coincided with similar endeavors that other institutions such as IRCAM in Paris or Stanford University were conducting in the same area, facing very similar technical, creative and experimental challenges [1].

By 1978, Asuar already had an outstanding career as electroacoustic music composer since 1959 when he composed *Variaciones Espectrales*, the first piece in Chile created mainly with electronic sounds [3]. However, during the 70's his interest in using digital technologies not only to synthetize new sounds but also to process and generate music scores, led him to undertake several projects related to computer-assisted composition.

During this period, he premiered *Formas I* (1971), the first orchestral score composed algorithmically in Chile using the IBM 360 computer [3]. Also, during this decade, he founded the Research group in sound technologies at the University of Chile, and the computer-assisted music albums *El computador Virtuoso* (1973) and *Así habló el computador* (1978) were published.

2.1 "A system to create music with a microcomputer"

Asuar devoted the last years of the 70's to the birth of the Comdasuar, which description was -according to himself- a computer with an "INTEL 8080 microprocessor. Its main musical features are the possibility to reproduce any musical score automatically, without the need of a human execution. In addition to that, the instrument can develop heuristic programs and propose musical ideas, based on sound probabilities or musical games." [1]

The computer used a text-based music writing system that allowed to build music sequences from one to six individual voices. This mechanism also allowed to edit, manipulate, sequence and even store (on cassette tape) musical data that the Comdasuar software could compile and convert into analogue signals throughout a six-voice polyphonic synthesizer.

The sound was by using a quartz oscillator along with a variety of analog effects: "a white noise generator, a rose noise generator, two ring modulators, two tremolo generators, two generators of functions oscillating at low frequency with the goal of generating sinusoidal, triangle, and square voltages" [2].

Considering the historical context, it is worth mentioning that the design of this computer anticipated the MIDI technology developed during the 80s, so it is perfectly possible to assert that "no other equivalent device existed in the world at that time" [3].

These features not only demonstrate an admirable engineering and imaginative capacity on behalf of Asuar in terms of musical creativity, but also introduces a discussion regarding the role of technology within creative processes and composition affairs.

3. GENERAL FEATURES

Since the unfortunate fact that the original algorithm is stored in the only existing copy of this machine -which is in a state of deterioration-, any attempt to investigate and "recreate" the Comdasuar software would require an interdisciplinary approach to analyze sources and recreate musical algorithms.
However, before attempting to emulate the Comdasuar software we must describe the main functionalities of this machine.

3.1 Asuar musical syntax

The Asuar musical syntax (AMS) is a text-based notation to describe melodic sequences as a list of pitch-duration pairs:
A specific note can be written by indicating an octave from 1 to 10, followed by a note name and optional alterations (see Table 1). If no octave is indicated, the last one will be used instead.

Octave	Pitch	Alteration (optional)
1 to 10 (center = 4)	A	R (3/4 tone down)
	B	W (flat)
	C	V (1/4 tone down)
	D	Q (natural)
	E	U (1/4 tone up)
	F	S (sharp)
	G	T (3/4 tone up)
	R (rest)	

Table 1. Pitch notation in AMS.

When it comes to rhythms, each duration is associated to a specific character based on their respective names in Spanish. As table 2 explains:

Rhythm	Code
𝅜	L
𝅝	R
𝅗𝅥	B
♩	N
♪	C
𝅘𝅥𝅯	S
𝅘𝅥𝅰	M
. (dot)	P

Table 2. Rhythmic notation in AMS.

Additionally, two (or more) durations can be added together by concatenation; for instance, "NC" will be equal to a quarter note plus an eighth note. That same duration can be notated as "NP" (dotted quarter note) or even "CCC".
The AMS includes three kinds of irrational rhythms: triplets (3), quintuplets (5) and septuplets (7). The number associated with each group must be indicated before the rhythmic character; for example, "3C" will be equal to an eighth note triplet, "5S" will be a sixteenth-note quintuplet, and so. By adding a "0" before the rhythm, irrational groups will not be applied (default).
It is crucial to understand that in AMS there are some repeated characters to define different elements; for instance, "C" stands for a specific pitch but can also indicate an eighth note. Therefore, the order of the elements within the list will be decisive for the compiler to clearly differentiate pitches from durations and mode changes.

3.2 Input modes

It is somewhat common to find repeated elements in musical structures of any kind. In order to avoid redundancies, Asuar designed a specific syntax to optimize and facilitate melodic notation.
As previously described, melodies are notated as a list of pitch-duration pair. This mode is defined as J0 mode according to the AMS and is enabled by default.
In this mode the "/" character can be used to indicate a repetition of either the last pitch or rhythm. The figure 1 shows an example of a well-known piece of melody.

```
5D N
4G C
A /
B /
5C /
D N
4G /
/ /
```

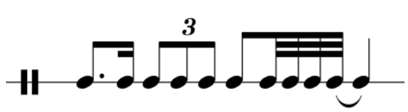
Figure 1. The beginning of Bach G major minuet in AMS (pitch-duration list). J0 mode is active by default and repeated notes and/or durations are notated with "/".

J1 mode is defined as "constant duration", which means that the rhythm notated after the J1 code will be repeated until another mode is executed. This mode is very useful inasmuch as it makes easier to notate a melody with a specific rhythm but different pitches.
Consequently, J2 mode is defined as "constant pitch". In this case, the pitch indicated after the J2 will be repeated until another mode is set. This mode is great to write rhythmic sequences and percussion lines.
As we can see in the examples shown in figure 2, these input modes make melodic notation considerably more efficient and clearer in cases of redundancies. In both situations we declare a constant value -either note or duration- and then list the variable element.

Constant duration:

```
J1 C
R
4D
E
F
G
A
BW
CS
BW
A
G
F
E
F
```

Constant note:




```
J2 4B
CP
S
3C
3C
3C
0C
F
F
F
FN
```

Figure 2. J1 and J2 examples in AMS respectively. In the second example, the is pitch notated as a central B but we can consider this sequence as a percussion line.

On the other hand, J4 and J5 modes were conceived to indicate different kinds of melodic repetitions:
J4 mode indication must be followed by the number of repetitions desired. All the notes listed between the J4 code and the next J mode will be repeated. This mode makes easier to notate tremolos and trills.
J5 mode is a function that copies a specific fragment of the melody and appends it at the end of the sequence, hence it requires two numbers after indicating the mode: the index of both the beginning and ending notes.
A melody that employs J4 or J5 modes can also include J1 or J2 within repetitions. For instance, we could use constant pitch mode (J2) and later declare a J4 mode to repeat the notes from that point to the next J mode declaration.

3.3 Basic heuristics

Asuar used the term heuristics to define a set of mathematical and stochastic processes that the Comdasuar could carry out based on melodic sequences stored in the memory (known as Voice Bank).
Some of these processes include basic modifications of the melodic sequence such as transposition, retrogradation and inversion.
In addition to that, Asuar also defines some more complex musical techniques such as pitch and rhythmic transmutations:
Pitch transmutation requires at least two sequences allocated in the Voice Bank. This process consists in extracting the pitch list from melody A and apply it to melody B so that rhythms will remain the same, but notes will be copied from the other voice [1].
Duration transmutation is the same process but based on rhythms instead of notes. In both cases the result is a third sequence that takes elements from initial melodies.
Also, the Comdasuar algorithm includes stochastic programs that can be used to manipulate or generate new material. According to Asuar descriptions, it is possible to apply some randomness to one (or many) parameters to a melodic sequence such as register, pitches, durations, harmony and sound texture [1].
These heuristics can be combined consecutively in order to build a diverse palette of music material that can be later organized on a timeline.

4. THE EMULATOR

The features previously described can be easily recreated using modern programming languages. Hence, the main purpose of this project is to investigate the prospects of the original Comdasuar and develop a software that emulates its functionalities and features based on the mentioned sources; so that it can be published online.
In order to do so, we followed these steps:
- Analyzing -from the perspective of musical theory and composition- the procedures described and exemplified in the available sources related to the original Comdasuar.
- Designing a software architecture (algorithm) that implements the procedures previously described.
- To develop a software that allows to emulate the procedures described, ensuring that the results of the emulator are consistent with the examples taken from the sources.

4.1 Restoring the source code

The Comdasuar emulator is being developed using Cycling'74 Max as the main framework. This well-known software has become a sort of *lingua franca* for computer-music composers, sound designers and media artists. Thanks to its graphical coding interface, creating simple apps (known as "patches") can be done without the need of using complex programming languages.
However, in this particular case, the use of Max is limited to the user interface and in/out operations (import and export files and media); whereas the core of the Comdasuar algorithm is completely coded in JavaScript. We call this collection of code modules Comdasuar.js Engine and is meant to be portable to any other environment that supports JavaScript (such as web browsers and mobile applications).
In an attempt to resemble the original architecture of the software, the source code of the Comdasuar.js Engine[1] includes different modules (classes) that work together depending on the needs of the user. The main modules are:

a) AMS parser: This module is programmed to compile melodic sequences written in AMS. Basically, it converts text into a set of lists related to durations, MIDI-notes and onsets.
b) Sequence object: The lists returned by the parser are used to create an instance of a Sequence class. This object contains all the melodic information following MIDI conventions for pitches, durations, onsets and velocities.
c) Voice Bank: it works as a database for Sequence objects. Sequences compiled can be stored, modified, processed and deleted from the Voice Bank. The emulator uses this module to save the bank in JSON format for later use.
d) Heuristics: this module features a variety of functions that can be applied to one (or more) sequences stored in the Voice Bank. Some of these musical procedures were described before in this paper.

Even when JV Asuar described in detail some of the features of the Comdasuar by addressing theoretical definitions along with musical examples; it is important to mention that from the original 26 sub-programs only a few are properly documented.
Therefore, the Comdasuar.js Engine features some extra modules that intend to recreate musical processes that JV Asuar mentioned but never explained in full detail.

4.2 The creative possibilities of the Comdasuar Emulator

Based on the examples previously shown, we can infer that the Comdasuar emulator can be a handy tool when composing. With only a few sequences we are able to build a virtually endless melodic palette that we can use to elaborate musical structures or maquettes.
Cantus II, a piece for flute and live electronics & visuals that I composed in 2018 was structured following this workflow: a small bank of melodic sequences from which new material comes out as a result of heuristic processes. Each section of the score introduces different heuristics applied to the same melodic material and, therefore, they suggest a constant development of the main musical idea throughout the piece.
This first approach to the Comdasuar Emulator as an assisted-composition media was undoubtedly fruitful, moreover if we consider that the piece features real time visuals and electronics; both conceived as an extension of the performer. However, this approach to the Emulator is based on a deferred time use of its potential.
In order to explore the creative possibilities of the Comdasuar Emulator we have developed an extra module called Sequence Generator that can carry out almost all heuristics in real time.
This module takes a set of sequences and performs different kinds processes (such as tone/duration transmutations, retrogradations and inversions) and, instead of storing this new melodic material in the Voice Bank, it reproduces it through a virtual MIDI out port called "Comdasuar-out".
By using this protocol, the Sequence Generator can "play" MIDI instruments such as external synthesizers or even VSTi inside a DAW. These parameters controlled by the user are related to the heuristics associated and -in addition to that- random values may be applied so that every new melodic sequence is unique.
We thought it could be interesting to try this module along with Ableton Live to create real time generated music and use a streaming service to broadcast the result of this experiment, that fits within the domain of Generative Music.
During the last months we have been broadcasting sessions of the "Comdasuar Generator" via YouTube[2]. In these videos the Emulator not only emits MIDI messages to virtual instruments in Ableton Live, but also determines different settings for audio effects according to preconfigured presets. The same material is constantly processed but never repeated literally, even when each session last half an hour at least.

4.3 The Comdasuar Emulator as a teaching tool

The Sequence Generator is a real time implementation of another module that also employs the Comdasuar.js Engine, we call it Solfeggio Generator; a program that creates melodic sequences and displays the result as a score by using traditional music notation[3].
Instead of a regular Voice Bank, this module uses a simpler data structure that we call Sets. The main difference

1 github.com/diegodelaFuenteCuraqueo/Emulador_COMDASUAR
2 https://youtu.be/Q2Nm_63JhMg
3 The interface of this module was possible thanks to the Bach library for Max: https://www.bachproject.net/

between them is that a Set stores small fragments of data and not complex melodic sequences.

The Solfeggio Generator requires two Sets: pitches (or scale) and rhythmic motifs (or cells). The sequence produced is a combination of randomly chosen elements from both lists, and this process is repeated until a specific number of bars is filled-out. Some of the parameters required to carry out this process are type of clef (G, F, C or percussion), time signature, number of bars and maximum melodic interval.

It is noteworthy that even when this module was not part of the original Comdasuar algorithm, it is strongly influenced by what JV Asuar described as music games. For example, the piece *Así habló el computador* (1978) was essentially created with this technique of combining elements from pitch and duration groups. The composer said that he obtained very interesting results from this heuristic when mixing elements from different music styles like popular songs, anthems or even serialism [1].

Therefore, the creative possibilities of this module are out of question. However -and considering the well-known challenges that the year 2020 brought to the world-, we were very curious about the possibility of using this tool as a sight-reading app for young musicians.

In consequence, this module was designed to replace music books and sheets during online classes. Each student was encouraged to sight-read a 4-bar music score tailored to his specific weaknesses regarding rhythm and/or pitches. When the student reaches the last note, a new score will be generated so he must keep the tempo and read *da capo*. The difficulty can be progressively incremented, and new notes/durations can be added to their respective sets.

This sort of musical game proved to be a great resource for music theory and sight-reading in online education. A small audiovisual record of the use of this tool in music theory classes is available online on the aforementioned platform[4].

5. MEDIA ARCHEAOLOGY

5.1 The Comdasuar as Cultural Software

The prospects of the machine described by JV Asuar show the dual nature of this piece of technology: on the one hand, we have the analog synthesizer (the hardware) and on the other hand, we have the computer algorithm explained above that includes 26 sub-programs to create, modify and generate musical elements (the software). From the combination of these elements emerges a hybrid system that combines score synthesis with sound synthesis [4], in which the computer medium becomes a cornerstone.

From that perspective, more than an electronic musical instrument, the Comdasuar can be understood as a whole computational meta-media that encompasses other analog-digital media [4], allowing to reconfigure their respective languages as a consequence of this hybridization. In words of Schumacher, the Comdasuar was not only a musical instrument; it was a true electro-acoustic laboratory, as would not be seen until several years later [3].

As we explained, the "creative" capacity of heuristics is based on the datafication of musical elements arranged in lists. These arrays are allocated in the memory in order to create new musical models generated from mathematical and random processes carried out by the computer. The concept of 'musical model' is a keystone in JV Asuar's creative philosophy regarding computer-assisted composition. In his own words:

"If we have been able to elaborate a program at a creative level to synthesize a certain musical style, we can connect it to another performance program, which in an intermediate phase will translate the musical structure and notation into magnitudes of the different sound parameters. In this way, it is perfectly possible to imagine the existence of computer programs that can be commanded externally in such a way that each musical relationship can be varied systematically." [7]

JV Asuar's vision of music computing, as well as the role of the computer in creative work, refers us to what in media theory (more specifically in the area of Software Studies) is called "cultural software", that is, a computer medium that offers the possibility of creating, exchanging and accessing cultural artifacts that contain representations, ideas, beliefs and aesthetic values; as well as participating in interactive cultural experiences [6]. An idea that forty years ago was not as evident as it is today.

Moreover, some of JV Asuar's reflections seem to anticipate -once again- the discussion about mediality both in the artistic context and culture in its broader sense: "The man-machine society could be defined, then, by giving to the machine everything that is mathematical calculation and leaving man the role to breathe spirit into matter, the capacity of communication, the computer would prepare matter and form; the composer would give it life and communication" [7].

These reflections relate to what Manovich would later define as "software society" and "software culture", emphasizing the prominent role that the software has acquired within the process of creating material elements and immaterial structures that, as a whole, allow to define 'culture' nowadays. [5]

If the Comdasuar was one of the first computers to make music, then it was also the first example of cultural software and media hybridization oriented to create music in Chile. This historical-cultural milestone had no major repercussions in its time, even more so considering that artistic development was violently interrupted by the military dictatorship in 1973. Sadly, from 1983 to the mid-1990s almost no electroacoustic works were produced in Chile [3].

6. CONCLUSIONS

It's been more than forty years since the Comdasuar was created and many doubts regarding its functionalities and

[4] https://drive.google.com/drive/folders/119bNkKWcu5GIWZ9R9zaSFRuucx4E11b0?usp=sharing

internal processes remain. Sadly enough, it is possible that some of those questions will never receive a proper answer because the original source code still inaccessible and its author passed away in 2017.
Nevertheless, the strong significance that such a unique endeavor implies for Chilean electroacoustic music history pushed us to formulate interdisciplinary methods to analyze and recreate part of the original machine.
Even when this first version of the Comdasuar Emulator will not include a Synthesizer like the original one, this software will allow us not only to understand some of the technical features, but also give hints about the way in which its creator conceived music composition and the prominent role of new media within this process.
The Comdasuar Emulator project is currently in testing phase and we hope to release the first version on GitHub this year.

7. REFERENCES

[1] J. V. Asuar. “Un sistema para hacer música con un Microcomputador”, Revista Musical Chilena, vol. 34, no. 151, pp. 5-28, 1980.

[2] M. A. Fumarola, “Report of the COMDASUAR: A Significant And Unknown Chilean Contribution In The History Of Computer Music”, Proceedings of the 1998 International Computer Music Conference, Ann Arbor, 1998.

[3] F. Schumacher. Música electroacústica en Chile. 50 Años. 2005.

[4] M. Supper. Música electrónica y música con ordenador: historia, estética, métodos, sistemas. Madrid: Alianza Editorial, 2012.

[5] L. Manovich. El software toma el mando. Barcelona, 2012.

[6] L. Manovich. El lenguaje de los nuevos medios de comunicación: la imagen en la era digital. Barcelona: Paidós Ibérica, 2005.

[7] J. V. Asuar, “Música con Computadores ¿cómo hacerlo?”, Revista Musical Chilena, vol. 26, no. 118, pp. 36-66, 1972.