Expert Systems With Applications 57 (2016) 345–355

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

MUSIC-MAS: Modeling a harmonic composition system with virtual organizations to assist novice composers

Maria Navarro[a,*], Juan Manuel Corchado[a,1], Yves Demazeau[b,2]

[a] *University of Salamanca, c/ Espejo, s/n, Salamanca 37007, Spain*
[b] *Laboratoire d'Informatique de Grenoble, 110 av. de la Chimie, Domaine Universitaire de Saint-Martin-d'Hres, Grenoble cx 9 G38041, France*

ARTICLE INFO

Article history:
Received 17 December 2014
Revised 30 January 2016
Accepted 31 January 2016
Available online 5 February 2016

Keywords:
Harmony
Consonance
Tension
Multi-agent system
Virtual organization
Harmony Search Algorithm

ABSTRACT

Many music students today experience difficulties in composing melodies without a prior harmonical guide. While harmony can be helpful in creating a melody the generation of harmony is challenging due to the many factors that must be taken into account, such as style, harmonic functions, musical consonance or aesthetics. Although various solutions have been proposed in the past, our study employs a different expert solution based on virtual organizations to make musical harmonies, which can assist novice improvisers and/or composers. The virtual organizations are implemented with Multi-Agent System (MAS) using PANGEA (Platform for Automatic coNstruction of orGanizations of intElligent Agents), a platform to develop different multiagent systems. The main goal is to simulate an expert multiagent system that can compose harmony following specific rules. To do so, the Harmony Search Algorithm is implemented as the main behavior of the composer agent, and adapted to a Belief-Desire-Intention architecture. The application of a VO has not been previously used in the development of this kind of expert system in music. We measured the quality of the music obtained, by minimizing a mathematical function. Additionally, we developed an evaluation test that positively validates the musical results from the perspective of consonance and usefulness of the composers.

1. Introduction

Interest in computational creativity is on the rise in the scientific community. Although this interest is recent, there are many algorithms, schemas and procedures to develop such an intelligent machine, capable of creating new ideas or new artistic compositions.

In the musical field, there are also many music students, and even musicians, who experience difficulties in composing melodies with their own instrument, and the may find it difficult to practice by themselves. For this reason, the system presented in this paper is designed to assist music students in improving their abilities without any external help.

The main contribution of this paper is to demonstrate how a new approach based on an agent framework can build a proper music generation system to assist novice composers. This will be accomplished by building an expert approach of harmonization useful for novice students of composition. A multiagent system based on virtual organizations is proposed in order to make a scalable and flexible expert system. This permits making changes in the problem specification, such as changing music style or adding new rules, without changing the structural composition. Only the agent behavior is modified, which is why the Belief-Desire-Intention (BDI) agent architecture was specifically chosen. It is also presented here a brief review of reviews several algorithms and methods used in harmony composition to highlight the contributions made by this system.

According with Delgado, Fajardo, and Molina-Solana (2009), due to the difficulty to evaluate music generation, a listening test may be needed to validate our results. At the same time, López-Ortega and López-Popa (2012) consider the evaluation must be a mathematical evaluation. Thus, we will evaluate the results by considering two types of criteria. To begin, we will consider mathematical criteria, which involve an optimization function. This function encloses constraint rules that are evaluated in the chord proposed as a candidate for our composition. A small value of this function for one chord studied means a good chord to incorporate to our composition. In fact, a threshold is established, so that if the chord exceeds this value when it is measured with the optimization function, it is dismissed, and the process starts again with a

* Corresponding author. Tel.: +34 923294500x5480.
E-mail addresses: maria90@usal.es (M. Navarro), corchado@usal.es (J.M. Corchado), Yves.Demazeau@imag.fr (Y. Demazeau).
[1] Tel.: +34 923294430.
[2] Tel.: +33 476514643.

http://dx.doi.org/10.1016/j.eswa.2016.01.058
0957-4174/

new chord that replaces the rejected chord. These rules will be detailed in Section 3.

Additionally, the concept of consonance is studied as a musical concept necessary to evaluate the system. This evaluation is made by musical experts as well as non-expert listeners.

A memory to store the different chords obtained is designed, imitating the memory used when a musician is composing. This Harmony Memory (HM) has a limited size due to the constraints of human memory and memory space. The new harmony will be included in the HM if it is better than the worst chord stored.

It will also be interesting to determine whether this system helps composers make their melodies or to improvise a melody by simply listening to the harmonies. As the user determines the number of bars or chords, the algorithm will create a fixed number of chords; however, the algorithm will be active in a loop until the stopping criterion (maximum number of improvisations) is reached, and will choose the best harmony among the possibilities.

The paper is structured as follows. Section 2 contains a brief review about the concept and definition of creativity, algorithms in music composition and artifacts made with these algorithms. The use of multiagent systems in computational creativity is also discussed and several examples are proposed. We will also introduce basic concepts of a Multi-Agent System (MAS) with a BDI architecture, and based on virtual organizations.

Section 3 presents our model with our particular solution, which attempts to solve the problem of harmony composition with an unknown melody, and demonstrate how Virtual Organizations (VO) can help to improve this system. Section 4 shows some results of the system, and proposes new lines of improvement.

2. Domain of knowledge

Two disciplines interact in this paper: artificial agents and music. We will begin by presenting a general and brief review about creative development, ideas and artifacts made. The first subsection introduces a brief background about computational creativity. The second subsection presents general information about composition algorithms, in which Harmony Search (HS) is discussed in greater detail, as it is the algorithm that was selected to solve the problem. The third subsection explains concepts about MAS and VO, whereas the last section includes a brief explanation about the background of agents, emphasizing the relationship between MAS and Creativity.

2.1. Computational creativity

Creativity is considered to be an essential component of human intelligence. Consequently, in attempting to answer the question of whether computers can think, it is only natural to ask whether computers can think creatively.

Some Artificial Intelligence researchers have tried to simulate creativity with computers. Among the most impressive programs developed are AARON (Cohen, 1995), a painting program that produces both abstract and lifelike works using a small robotic turtle, combined with several drawing strategies, or the Ebcioglus CHORAL system (Ebcioglu, 1988), which was able to produce chorale harmonizations rather similar to those of J.S. Bach. After these experiments, and partially due to the lack of conceptual and theoretical consensus, there have been scientists interested in exposing new theories about creativity and concepts related to it. One of the most important is Boden, who proposed a framework and a creativity view (Boden, 1987) which continues to have philosophical impact in computational creativity (Colton, Wiggins et al., 2012).

Boden defines creativity as an ability to conceive new, surprising and valuable ideas and artifacts. These three elements must exist in all creative phenomena. Another remarkable theorist, Csikszentmihalyi (1997), defends the argument that creativity consists of three main parts: the domain (a set of symbolic rules and procedures); the field, which includes all the individuals acting in the domain; and the individual.

Creativity concepts in AI, such as imagination, surprise, novelty or emergence, make it possible to conceive psychology in a new way, in order to build and test hypotheses about structures and processes involved in the mind. Some of these concepts have been considered controversial due to their abstract nature. In particular, emergence is a concept that has multiple definitions depending on the field studied, the theory in which it appears, or the author (Deguet, Demazeau, & Magnin, 2006). Despite this, emergence is a concept used to value complex systems; in our particular case, creative systems are considered as complex systems (Deguet et al., 2006).

With regards to the mental process that machines can simulate to be creative, Boden is interested in running tests to determine whether computers can conceive ideas that are considered or appear to be creative. In “AI and natural man”, Boden (1987) considers AI to be the science of thinking and action, which suggests that computers are tools that provide themselves with a “human-like” intelligence. Along these lines, Lösch, Dugdale, and Demazeau (2009) identify aspects about creativity in individuals, as motivation, externalization, inspiration, etc., to define requirements and functionalities of a model capable of enhancing the creative abilities of the user in design tasks.

2.2. Musical composition algorithms

Music is considered to be an interesting research area in a variety of research fields because it deals with a human activity that is both intellectual and emotional. It is a universal language quite different from spoken language.

For example, telecommunications researchers are interested in music software because since music is considered a form of information and they need to know general characteristics of information and how it can be manipulated for its broader dissemination. Philosophers are interested in music software because the ability to specify musical compositions (i.e. “to compose”) at a higher level than note-by-note would bring them one step closer to reaching a direct expression of musical ideas.

Software engineers also find formidable challenges in areas such as music composition; the simulation of this complex activity requires expertise in algorithm design, expert systems, optimization, and other related software engineering disciplines. Designing an algorithm to compose music has no simple, mechanical test for success.

Initially, grammar-based systems were widely used in composition tasks. By thinking that music follows grammatical rules, many computer composers modeled music relationships as grammatical structures, representing musical structures (Roads & Wieneke, 1979). In fact, Holtzman (1981) creates a musical grammar that generates harp solos based on the physical limitations imposed on harp performers. Cope (1987) derives grammar from the linguistic principles of haiku to generate music in a particular style. Although grammar can produce a natural sound, the tasks corresponding to deciding the aspects of a musical structure that should be represented are often difficult and ad-hoc (Marsden, 2000). Nowadays, there are many other algorithms that attempt to compose music, some of which are called live algorithms (Bown, 2011).

One of the most successful algorithms is the Markov Models (Eigenfeldt & Pasquier, 2013). There are also algorithms that use lyrics as a variable in their compositions, as for example Monteith, Martinez, and Ventura (2012) or genetic algorithms (Pereira, Machado, & Cardoso, 1998). In this sense, one interesting study to note is that of Pachet (2003). Pachet proposes a system referred to as the Continuator, which is able to build operational representations of musical styles in a real time context. The model is based on Markov Models to deal with rhythm, pitches and imprecisions. The resulting system is able to learn and generate music in any style and also makes it possible to create new modes of musical collaborative playing (Pachet, 2002).

Regarding with bio-inspired and machine learning algorithms, Hoover, Szerlip, and Stanley (2011) focused on evolving a single monophonic accompaniment for a multipart MIDI. These accompaniments are generated through two functions, one each for pitch and rhythm, which are represented as a compositional pattern producing network (CPPN), a special type of artificial neural network (ANN). CPPNs can evolve to assume an arbitrary topology wherein each neuron is assigned one of several activation functions. Mocholi, Martinez, Jaen, and Catala (2012) addressed the problem of music playlist generation by using a multicriteria ant colony, and López-Ortega and López-Popa (2012) present a suite to assist in the creation of musical pieces, whose foundation lies on fractals, fuzzy logic and expert systems. Another good example is the recognition system for western music made by Mostafa and Billor (2009), based on machine learning algorithms.

Recently, Velardo and Vallati (2014) propose a memetic model for music composition, which considers both psychological and social levels. It permits people to experiment music generation but it is not focused on classical music generation. The Music Room (Morreale, De Angeli, Masu, Rota, & Conci, 2014) is an interactive system where a couple composes music by moving in the space. This is an interesting work, but it is not focused on classical music. AutoRhythmGuitar (McVicar, Fukayama, & Goto, 2014) permits to compose rhythms by using some input parameters such as chords and melody. Thus, the harmony composition is put aside in this work. Dubnov and Assayag (2013) also create a computer-aided composition using MIDI files as inputs to create a new melody. In this last work, the harmony generation is also out of the scope. López-Ortega and López-Popa (2012) present an expert system to assist composing musical pieces. This work follows fractal rules; thus it is not able to compose tonal harmony. Additionally, it is not oriented to students that want to learn tonal music composition. Finally, Navarro, Caetano, Bernardes, de Castro, and Corchado (2015) present a system able to create chord progressions based on an AIS that also helps novice composers. However, it is based on an evolutionary algorithm, which does not provide the flexibility of mulatiagent systems to adapt the system to different contexts.

Consequently, the background given about creative systems demonstrates that the concept of creativity can be used as a phenomenon to develop a potential expert system. However, although creative systems propose many different ways to create a composition, they are not specifically designed for novice composers, as is our proposal.

2.2.1. Harmony Search Algorithm

Music improvisation wants to produce an ideal state determined by aesthetic parameters, such as for example, consonance or sound balance.

Similarly, algorithmic optimization seeks to produce an ideal state as determined by a function referred to as the objective function. In the case of improvisation, the aesthetic result is produced by a set of pitches (harmony) played by different instruments. On the other hand, the optimization function is performed by a set (vector) of values within all the decision variables (Geem, Kim, & Loganathan, 2001). Fig. 1 shows the analogy between music improvisation and optimization. Each musician (saxophonist, double bassist, and guitarist) is matched with each decision variable (x_1, x_2, and x_3). In addition, the range of each music instrument (saxophone = Do, Re, Mi, Fa, Sol; double bass = Do, Re, Sol, La, Ti; and guitar = Do, Mi, Fa, Sol, La) is matched with the range of each variable value (x_1 = 1, 2, 3, 4, 5; x_2 = 1, 2, 5, 6, 7; and x_3 = 1, 3, 4, 5, 6).

Therefore, if the saxophonist plays the note Re, the double bassist plays Ti, the guitarist plays Mi, and their notes make a new harmony (Re, Ti, Mi). If this new harmony is consonant (aesthetically pleasing), the harmony is kept in the musicians memory. The new solution vector (2, 7, 3) generated in the optimization process is also kept in the computer memory if it is good in terms of objective function (Geem & Choi, 2007; Geem et al., 2001). The procedure has five steps described in this section.

To begin, the optimization function, which evaluates the improvised harmony is defined as follows:

$$\text{Optimize } f(x) \tag{1}$$

$$\text{Subject to } x_i \in X_i, i = 1, 2, ..., K \tag{2}$$

where x is the set of each musical instrument (decision variable) x_i; X_i is the set of candidate pitches for each musical instrument, that is, $X_i = \{x_i(1), ..., x_i(K)\}$ where $x_i(1) < \cdots < x_i(K)$; and, finally, N is the number of voices.

This algorithm has a memory called Harmony Memory (HM) matrix filled with as many generated solution vectors such as HMS (harmony memory size).

To improvise a new harmony (solution vector $x = (x_1, ..., x_K)$), three rules must be followed:

- *Random selection:* The value of x is a random choice of any pitch within the established range.
- *Memory consideration:* The value of x is chosen from the HM, with a probability of HMCR (Harmony Memory Considering Rate).
- *Pitch adjustment:* The value of x is chosen from the HM and then modified with a probability of PAR by a random value following the next equation:

$$x_{new} = x_{old} + rand \in (0, 1) \times bw \tag{3}$$

where bw is an arbitrary distance bandwidth used to improve the performance of HS. The value of bw is an optimization problem itself.

Although the new harmony is built, it is necessary to consider the constraint rules that evaluate the resulting chord. If the new harmony does not fall within the constraints, it may still be used, but with a penalty, which in turn depends on the programmer. For example, the rule-violating harmony of parallel fifths was considered as a constraint in the application presented in this paper, but is not enough to remove this chord from the harmony composition; therefore, a threshold is established. If the chord exceeds this value, it is dismissed, and the process starts again with a new chord that replaces the rejected chord. Finally, if the new harmony vector, x, has a better value for the fitness function than the worst harmony in the HM, the new harmony is included in the HM and the existing worst harmony is excluded from the HM. This process is repeated until the stopping criterion (maximum number of improvisations) is reached.

The music generation techniques described here deal only with syntactic (i.e. form) considerations; the programs have no method for handling semantics. The semantics of music involves an immense wealth of intricately interrelated experiential, cultural, and historical information (the result of activities loosely called “experience”, “acculturation”, “learning”, etc.) that does not yet exist in any machine manipulable form. Semantics is well beyond the scope of this work. However, experience shows that having the correct form is often sufficient to produce pleasing music according to the success criteria presented in previous sections.

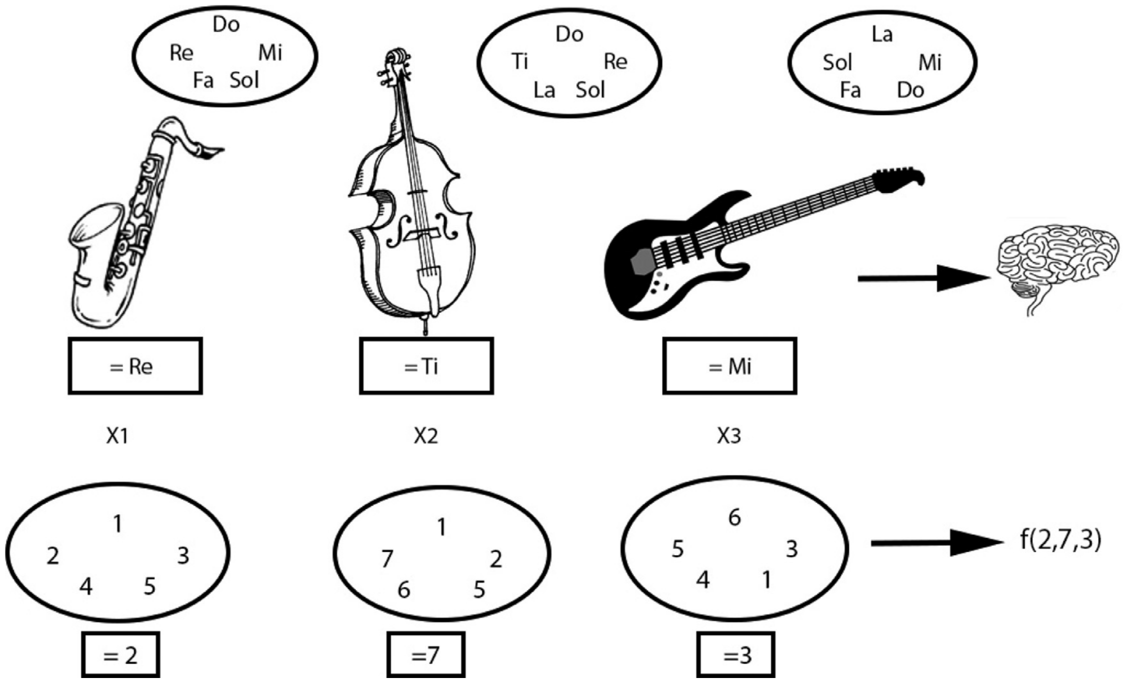


Fig. 1. A musical improvisation or composition with three instruments or voices is shown. The notes are translated into numbers to make calculus with the possible combinations and to optimize the mathematical function proposed.

2.3. Multiagent system

The concept of agent is present in various fields of study, such as psychology, computer science, sociology, medicine, economics, etc., with different points of view and behaviors in each one. In the field of Computer Science, the term agent is becoming more known and used in areas as diverse as the Internet, distributed artificial intelligence or human–computer (Corchado, Pavón, Corchado, & Castillo, 2004) interaction systems.

Nowadays, there are many multiagent frameworks, which help and facilitate working with agents (Galland, Gaud, Rodriguez, & Hilaire, 2010; Giret et al., 2010; Howden, Rönnquist, Hodgson, & Lucas, 2001; Hübner, 2007). General purpose was considered a major issue twenty years ago, but it is much less the case now, at a time where personal computers, devices, mobile phones and similar devices, have grown exponentially. The architecture required for harmony composition must be able to process tasks specific to harmony composition. This is why we have developed the case study over the PANGEA (Platform for Automatic coNstruction of orGanizations of intElligents Agents) platform (Zato et al., 2012). PANGEA has a dynamic and adaptable nature; it integrates new services for Harmony Search Algorithms or composition. The proposed MAS is based on virtual organizations (VO) (Boissier, Gâteau et al., 2007a; Dignum, 2004; Ferber, Gutknecht, & Michel, 2004), and incorporates social behaviors. An organization of agents is defined as a social entity composed of a specified number of members, who perform various tasks or functions, and which are structured according to a specific communication pattern and topology to achieve the overall objective of the organization, based on rules of behavior. The main factors are aspects or structure, functionality, dynamism, normalization and environment (Boissier, Hübner, & Sichman, 2007b; Ferber & Gutknecht, 1998; Hübner, Boissier, Kitio, & Ricci, 2010). The new concept of multi-agent systems and virtual organizations or societies has resulted in two different types of methods according to the applied process of development (Dignum, Meyer, Weigand, & Dignum, 2002; Ferber, 2003). Thus, if the process is guided by the organization of the system, we say that the methodology is oriented to the organization (organizational-oriented methodology). However, if the process is focused on the specification of individual agent actions, we say that the methodology is oriented to the agent (agent-oriented methodology).

This leads us to conclude that virtual organizations of agents are an ideal option to create and develop the open and heterogeneous systems such as those normally found in the composition process. The use of virtual organizations of agents facilitates the incorporation of new composition techniques to the system.

Agents and creativity are two disciplines that have interacted in several cases studies. For example, Martin, Jin, and Bown (2011), or SC-EUNE (Macedo & Cardoso, 2001), or Machado, Romero, Santos, Cardoso, and Manaris (2004). Even a new approach of creative agents, called motivational agents, was proposed and used to explore unknown environments (Macedo & Cardoso, 2004). Delgado et al. (2009) built Inmamusys, a music system based on agents and expert systems. Other examples are Lacomme, Demazeau, and Dugdale (2010), an artistic performance made by a multiagent system. Interaction also plays an important role in computational creativity, particularly interaction between computers and humans (as the generators or users of the computational system; Maher, 2012).

Eigenfeldt, Bown, and Carey (2015) describe the musebot and the musebot ensemble to explore collaborative methodologies based on a combination of agents and communities to create a collective composition, not using interactivity between the human and the machine. Herremans, Weisser, Sörensen, and Conklin (2015) create an structured music for bagana, and Ethiopian instrument, based on Markov Models. This proposal is not centered in general tonal harmony, and can be only applied to rhythm or melodic patrons in baganas. Herremans and Sörensen (2013) create a counterpoint music with a search algorithm. This work is quite similar to the one presented in this paper, as they use a search algorithm similar to the algorithm proposed here, but the harmony is not addressed, and it is not oriented to assist an external user. Kirke and Miranda (2015) propose a multi-agent system which generates melody pitch sequences. The agents have melodic intelligence and generate the pitches as a result of the artificial emotional influence and communication between agents, and the melody's hierarchical structure is a result of the emerging agent social structure. They use MAS but only for melodic purposes, not considering harmonic constraints.

The state of the art highlights MAS as an enabler technology to develop the present composer system because it seems to be extensively used within this context with successful results. VO is used here due to its flexibility to operate in an environment with changing elements such as the harmonic changes in a composition.

Among the diverse existing algorithms, Harmony Search (HS) was chosen as the most suitable algorithm to be applied into our VO, because of its suitability for this problem. Prior to our experiment, a VO had not been used before our experiment in the context of Music Generation. Thus, this combination of VO and the heuristic concepts to generate an expert chord generation system can be considered a novel approach. It is to note that the application of this expert system is composition assistance, thus, this system contributes to the community of novice composers who have difficulties in creating their own melodies. The following sections presents a description of the system, as well as the algorithm, and agents structure used to solve the problem.

3. Modeling the problem

This section is divided into several parts. The first part presents our specific case study, detailing the algorithm used for composition, and the constraints considered. The second shows how the VO-based MAS based is built and the advantages of using this sort of architecture. The last part presents some results of our approach and their evaluation from a mathematical as well as musical point of view.

3.1. Case study: classic harmony composition

This case study develops a system that improvises a harmony following the next classic harmony rules:

- **R1** – 8th and 5th parallels. Produced when the interval between the *i*-note and the *j*-note of chord *n* and the interval between the (*i*+1)-note and the (*j*+1)-note of the chord *n*+1 are both 5th or 8th.
- **R2** – Leading-note resolution. There is a rule that requires solving the leading-note in the tonic.
- **R3** – Voices crossing. An ideal harmony must avoid voice *i* being above the voice *j*, when $j = i + 1$.

The Harmony Search Algorithm is used to composing harmony. As is previously explained, the first step is to choose the optimization function. We chose to design a custom function as the optimization function to satisfy our own needs.

When a musician decides to compose there are some general rules to consider. These rules are essential to compose proper music. Those which we want to consider are:

- **R4** – Movements between *tension*. Each chord has a specific role that produces stability or instability, depending on the functions (tonic, dominant and subdominant). Chords corresponding to these functions are shown in Fig. 2. This is the *tension*, which permits the music to evolve in the composition. For this reason, our desire is a movement between chords, to avoid the music getting boring. So, in some way we have to penalty the repetition of the same function over time.
- **R5** – Avoid a large interval between two pitches in a chord. We are considering an accompaniment with only one instrument, e.g., a piano. This is important because if we have a big pitch in the same chord, the connection between all pitches can break.

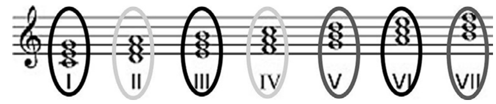


Fig. 2. Relationship between degrees in C Major. The chords surrounded by black circles are related to tonic function, the ones surrounded with a gray circle are related to dominant function and the ones surrounded with a white circle are related to subdominant function.

Table 1
Table showing the values of *iRank*(x).

Interval	*iRank* (interval rank)
3rd	1
6th	1.5
4th	2
5th	2.5
8th	3
Unisone	3.5
2nd, 7th	4

Table 2
Table showing the values of *Tension*(x)

Interval	Interval rank
Tonic	2
Dominant	3
Subdominant	1

- **R6** – Avoid a large interval between two pitches in the same voice. This rule allows building more "cantabile" melodies, in general.

With all of these constraints and rules, the next optimization equation is built to minimize:

$$\sum_{i=1}^{N}\sum_{j=1}^{3} Rank(x_{ij}) + \sum_{i=1}^{N}\sum_{j=1}^{3} Penalty(x_{ij}) \quad (4)$$

Where:

$$Rank(x_{ij}) = iRank(x_{ij}, x_{i(j-1)} + ln(Tension_i) + x_{ij} - x_{(i-1)j} \quad (5)$$

Constraints:

$$x_{(i-1)j} \equiv SI \Rightarrow x_{ij} \neq DO \quad (6)$$

$$x_{i(j-1)} < x_{ij} \quad (7)$$

$$Tension_{i-1} = 3 \Rightarrow Tension_i = 2 \vee Tension_i = 3 \quad (8)$$

$$x_{(i-1)j} - x_{(i-1)(j-1)} = x_{ij} - x_{i(j-1)} \neq 5 \vee 8 \quad (9)$$

In classical composition, intervals between pitches are essential. For this reason tension between chords are considered less important in the function with a logarithmic function to smooth its value. Values of *iRank*(x) and *Tension*(x) are shown in Tables 1 and 2, and *Penalty*(x) are shown in Eqs. (10)–(13).

$$x_{(i-1)j} \equiv SI \wedge x_{ij} \neq DO \Rightarrow Penalty(x_{ij}) = 5 \quad (10)$$

$$x_{i(j-1)} \geq x_{ij} \Rightarrow Penalty(x_{ij}) = 4 \quad (11)$$

$$Tension_{i-1} = 3 \wedge Tension_i = 1 \Rightarrow Penalty(x_{ij}) = 2 \quad (12)$$

$$x_{(i-1)j} - x_{(i-1)(j-1)} = x_{ij} - x_{i(j-1)} = 5 \vee 8 \Rightarrow Penalty(x_{ij}) = 3 \quad (13)$$

This case study considers only pitches in the center of the staff, between Do3 and Mi4, as is shown in Fig. 3.

The algorithm starts with an initialization of the Harmony Memory (HM) matrix. Three voices are considered, so the matrix

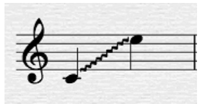


Fig. 3. Rank of decision variables. These are between Do3 and Mi4.

includes four columns, one for the value of the optimization function, and the others for each voice. The HM can be filled randomly or by the user or programmer, because it is stored in the repository.

We tested several PAR and HMCR. The system is able to compose on its own; however, if the PAR or the HMCR is too high, harmonies cannot adapt to classical standards. We have to find an equilibrium. Finally, the values chosen are 0.3 to PAR and 0.2 to HMCR.

The next section explains the structure and advantages of MAS based on Virtual Organizations (VO).

3.2. Agents and roles

Fig. 4 gives a schema of the workflow of the system. Upon considering information about music composition, and pondering which type of system to build, we have decided to use virtual organizations to develop our model. This model provides certain advantages. First of all, virtual organizations provide a certain number of roles that are easily replaceable by an agent, depending on the context. This permits the system to be very flexible. Additionally, to follow a methodology based on Virtual Organizations can give us a global vision about the problem, the model and the possible solutions.

When designing a virtual organization it is necessary to analyze the needs and expectations of the system. The result of this analysis will be the roles of the entities involved in the proposed system. In our case, the following roles were found:

- Composer role: This role creates harmonic music following their rules to achieve a goal (desire).
- Evaluator role: This role evaluates the result of the composer role and decides if it is good enough to present it to the user.
- Interface role: This role allows the user to interact with the system. It is responsible for collecting the ratings that the user can get, and for displaying the system results to the user.
- Data supplier role: This role is an agent that accesses and stores all or most of the information needed to manage the actions that govern this system.
- Control role: The agents that carry out this role will have overall control of the system. They analyze the structure and syntax of all messages in and out of the system.

However, in order to develop this virtual organization proposed and to adapt it to our requirements, it is necessary to implement a MAS. Each agent or group of agents in the system will perform a role in the virtual organizations. The agents corresponding to the composer and evaluator roles are particularly complex and are needed for a more complex structure. Fig. 5 shows a visual representation of the relation between the agents and the roles.

Among the different MAS architectures, we chose BDI agent architecture, for two reasons: firstly, it is the most common deliberative agents architecture and one of the most simple; and secondly, this structure is perfectly suited to our requirements. The BDI agent process involves two fundamental activities: (a) determining which goals should be achieved (deliberation) and (b) deciding how to reach these goals (planning). Both processes should be carried out by taking into account the limited resources of each agent.

The system is designed as follows:

- The goal of the composer agent is to minimize the value of the optimization function. This is her “desire”. To achieve this goal, she has to make some rules, in other words, to follow her “intentions”(that is the algorithm), starting from her “beliefs” or her initial stage. As we can see, the BDI architecture is completely suitable to the agent.
- The composer agent has as a goal to classify the chord made by the composer agent. This is her “desire”. To achieve this goal, she has to follow her “intentions”, starting with her “beliefs”.
- The rest of the agents are given communication, coordination and representation tasks.

The system was developed on PANGEA (Zato et al., 2012), which provided us with certain advantages. Due to service orientation, it is possible for the platform to include both a service provider agent and a consumer agent, emulating a client-server architecture. The ProviderAgent (a basic agent that provides a service) knows how to contact the web service, while the remaining agents know how to contact with the ProviderAgent due to their communication with the ServiceAgent, which contains information about the available

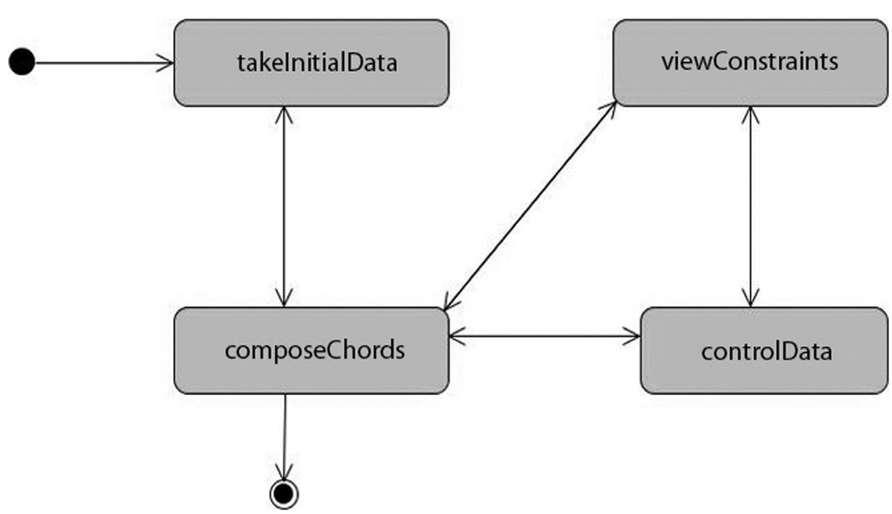


Fig. 4. State machine of the overall system behavior.

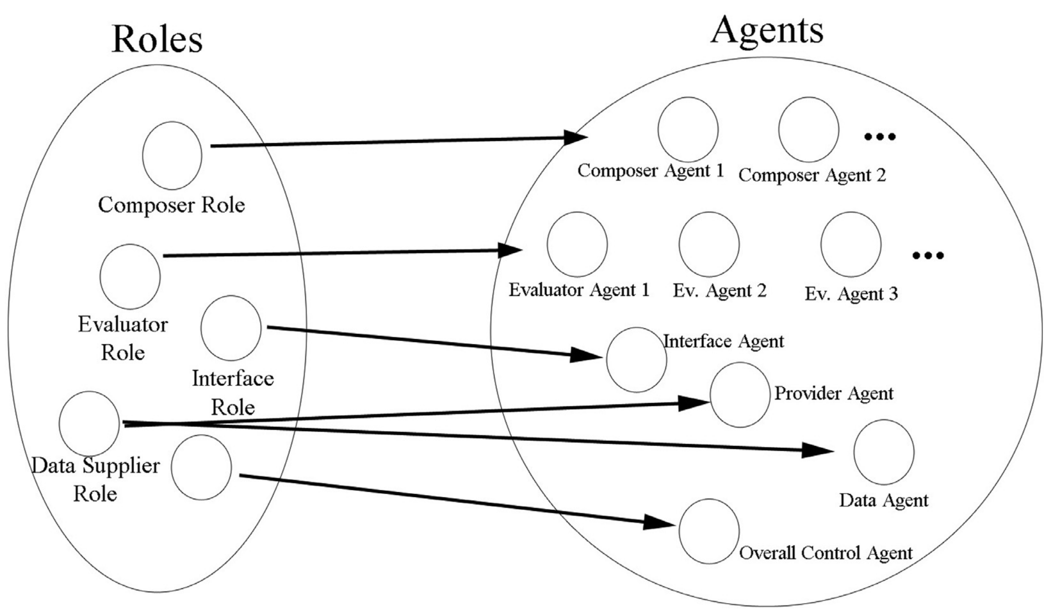


Fig. 5. Schema that shows how the roles are implemented by different agents. This architecture permits to build a flexible system with different composer or evaluator agents.

services. The schema in Fig. 6 shows how client agents are connected to model our problem.

Once the ClientAgent's request has been received, the ProviderAgent extracts the required parameters and establishes the contact. Once the contact is established, the results are sent to the client agent. Using Web Services also allows the platform to introduce the Service-Oriented Architecture (SOA) (Josuttis, 2007) into multiagent system systems.

As we can see in Fig. 6, the Interface Agent is able to interact with the environment and to pass the information to the system. Inside, there is a musical agent, capable of composing music following our HS algorithm. There is also an Evaluator agent which marks all the restrictions, studies them and gives different values for each constraint. The Control Agent evaluates the quality of the music composed by the musical agent, and decides whether to incorporate the new chord to HM, and validate all the data given.

This system has several advantages: first, we can change our musical agent in order to change the composition algorithm or behavior. We can even change an agent and replace it with a multiagent system capable of communicating to compose a new music. Additionally, we can change our Constraint Agent. This means that different styles can be composed with this system and we need only incorporate new behavior or update it to make jazz, rock, romantic, baroque or medieval music. We also have a database with classic styling features. The user can change these features and behaviors at any moment to permit or forbid parallel 5th or 8th, study the leading-note resolution, etc.

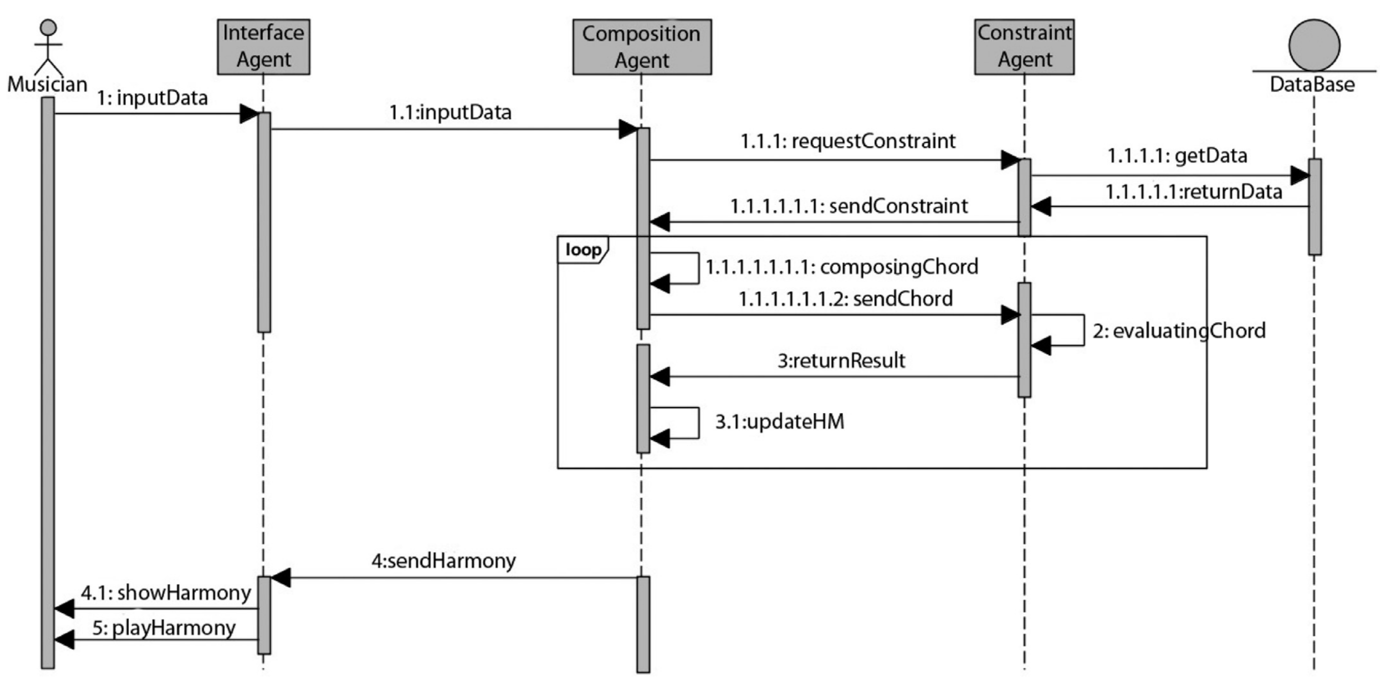


Fig. 6. A global view of multiagent system interactions among only client agents.

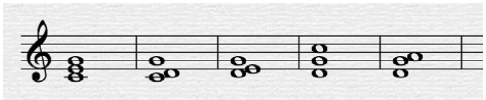


Fig. 7. Harmony achieved with 45 iterations.

3.3. Performance evaluation

The expectation is to go ahead and make a pseudo-intelligent system able to create new chords and new harmonic music, adjusting to the input rules. The fitness of the results is evaluated by studying the way the rules and constraints are followed. In other words, the more closely the rules are followed, the better the harmony will sound. The mathematical evaluation is to study the value of the optimization function as well as the number of the constraints that are violated.

After the first iterations, we did not obtain a proper chord line, as we can see in Fig. 7. The first chord is a perfect one for taking into account the intervals between the notes. Upon analyzing the transition between chord 1 and chord 2, we can see that the intervals are not quite perfect (between Do and Re there is a 2nd interval, which is considered dissonant). Between chord 2 and 3, the R3 is violated, as Do is becoming Mi, and the intervals again are not so perfect. Chord 4 has consonant intervals (although they might be better) but in the third voice rule R6 is violated (Sol becomes Do, and this is a big interval.) In the end, chord 5 is better for rule R6.

To study rule R4, we consider a chord to be a Dominant chord if it has the notes Re and Ti or the notes Sol and Ti. We have no Ti in any chord, so this rule is not evaluated in this sample.

However, the more iterations we performed, the better the results we obtained. We have a new line with 200 iterations, notably better than the previous one (see Fig. 8).

The first chord is a perfect chord if we keep into account the intervals between the notes. Analyzing chord 2, we can see that the intervals are almost as perfect as those of chord 1 (we have a 3rd interval and a 4th interval). Chord 3 is a chord with perfect consonance. Chord 4 has a consonant 4th interval and a dissonant 2nd interval. Finally, chord 5 is consonant with a 3rd and 4th interval.

Rules R3, R5 and R6 are followed throughout the experiment. To study rule R4, we consider a chord to be Dominant if it has the notes Re and Ti or the notes Sol and Ti. We have no Ti in any chord, so this rule is not evaluated in this sample.

This means that we have an evolutionary algorithm. This depends not only on the iterations we perform, but also on the parameters PAR or HMCR indicating the probability of making a random value for a pitch in a chord. If this value is very small, our system only imitates sounds stored in HM. However, the expectation is to enable a pseudo-intelligent system to create new chords and new harmonic music, adjusting to the input rules. The fitness of the results is evaluated by studying the way the rules and constraints are followed. In other words, the more the rules are followed, the better the harmony will sound. The mathematical evaluation is to study the value of the optimization function as well as the number of the constraints that are violated. But in music, mathematical and formal evaluations are not enough to decide if a system is good or bad.

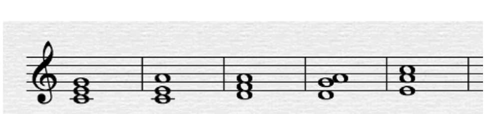


Fig. 8. Harmony achieved after 200 iterations.

There is also a qualitative form to evaluate the model. From a musical point of view, the concept of consonance should be considered. The definition of consonance has been based on experience, frequency, and both physical and psychological considerations. These include:

- Perception blend/fusion: perception of unity or tonal fusion between two notes.
- Frequency ratios: ratios between frequencies can determine the consonance of a sound. When the ratio is a simple number the sounds are more consonant than those that do not have a simple number as a ratio value (Pythagoras).
- Coincidence of partials: Consonance depends not only on the width of the interval between two notes (i.e., the musical tuning), but also on the combined spectral distribution and the sound quality of the notes.

According to these criteria, we can consider these consonance intervals (in order of consonance):

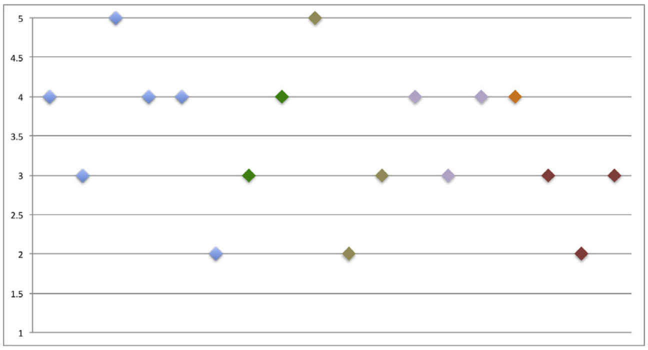
- Unisons and octaves
- Perfect fourths and perfect fifths
- Major thirds and minor sixths
- Minor thirds and major sixths

In Western music, dissonance is the quality of sounds that seems “unstable” and has a need to “resolve” to a “stable” sound called consonance. Both consonance and dissonance are words applied to harmony, chords, and intervals and, by extension, to melody, tonality, and even rhythm and meter. Although there are physical and neurological facts that are important to understanding the idea of dissonance, the definition of dissonance is culturally conditioned. Conventions of usage related to dissonance vary greatly among different musical styles, traditions, and cultures. Nevertheless, the basic ideas of dissonance, consonance, and resolution exist in some form in all musical traditions that have a concept of melody, harmony, or tonality. Dissonance, as the complement of consonance, may be defined as a non-coincidence of partials, lack of fusion or pattern matching, or as complexity.

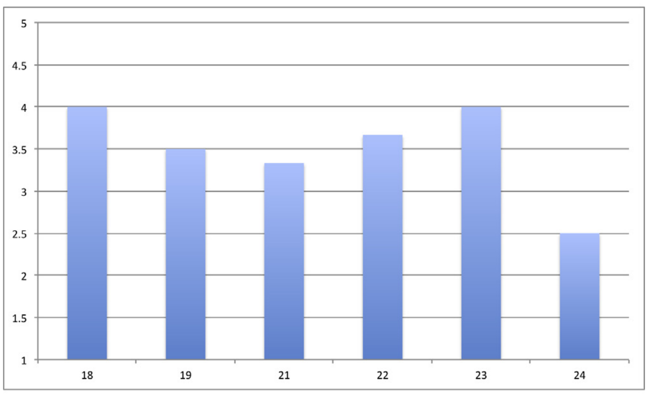
All music with a harmonic or tonal basis and perceived as harmonious, incorporates some degree of dissonance. The buildup and release of tension (dissonance and resolution) is partially responsible for what listeners perceive as beauty, emotion, and expressiveness in music. This dissonance and resolution is considered in the algorithm presented as the “tension” in the harmonic music.

Although it is a concept that provokes controversy and depends on the cultural tradition, consonance, in this case, would be a combination of notes that sound pleasant to most people when played at the same time. In particular, we consider a chord made by 3th, 5th, 6th or 4th intervals to be consonant, although there are some intervals more consonant than others (3th is more consonant that 4th). Consequently, this method of evaluation is a method based on acoustic perception, and therefore depends on the listener. Despite the condition of the listener, it is common for experts to evaluate with the same values. In fact, we conducted tests with two experts in classical music (composers) and two non-experts in classical music to evaluate both harmonies above. The evaluation criteria were: “completely dissonant”, “dissonant”, “a bit consonant”, “consonant” and “completely consonant”. Experts number 1 and number 2 evaluated the first harmony as “a bit consonant” and “dissonant”, while the non-experts evaluated it as “dissonant”. In the second harmony all four agreed to evaluate it as “consonant”.

A second experiment was carried out with 18 novice composers between 18 and 24 years old. All of them studies first year of Harmony in the Conservatory of Music, and they have some difficulties in creating their own first melodic compositions. They were provided with our system in their computers and make their first



(a) Plot that shows the rates that each composer gave to the compositions that the system made. Values above three are considered as positive rates.



(b) Plot that shows the mean rates that the composers gave to the overal system, grouped by age. Values above three are considered as positive rates.

Fig. 9. Plots with the evaluation of the novice composers that tested the expert system developed here. (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

composition by using the harmony created by our proposal. They made five different melodies with five different results given by the system. First of all, they have to evaluate if the harmony made is a proper classical harmony to work on it as a novice composers, from 1 (completely disagree) to 5 (completely agree). Additionally, they have to rate if their skills were improved after to use this tool during two weeks, from 1 (completely disagree) to 5 (completely agree).

The results obtained of this experiment are plotted in Fig. 9. Fig. 9a points are highlighted in different colors to know about the age of the composer that rated the system. Therefore, blue points are for 18 years old composers, green for 19, brown for 20, pink for 21, orange for 23 and red for 24. In general, we can see that 15 of the composers evaluated the system as a proper system to make classical music, whereas three of them deemed it not a proper one. Thus, it might be stated that the system makes a proper tonal harmony for the most part of the community. The age group with the most negative rates is the group of 24 years-old. This could mean the system is not useful enough to elder people. However, a further study of this experiment by age is out of the scope of the work.

Fig. 9b shows the mean improvement by age that they think they have done. As we can see, the group of age 24 are the group considered less influenced by the hypothetical benefits of our applications. Thus, we can state that our system would be useful for a large proportion of the community of novice composers.

Additionally, they were asked by the interactivity of the activity. All of them coincided that the interactivity and interface is good enough, but made some suggestions, such as to give some recommendations to the user about melody and to add rhythm to the harmony in order to help to decide the rhythm in the melody. Anyway, these results leads us to conclude that the system can build a nice classical harmony helpful for music students.

Table 3
Comparative table between two previous systems and MUSIC-MAS. System 1 means Kirke and Miranda (2015) work while System 2 means Navarro et al. (2015) work.

	MUSIC-MAS	System 1	System 2
Harmony generation	X		X
Melody generation		X	
Uses MAS	X	X	
Extensible system due to its design	X		
Uses classical rules	X		X
Uses emotions		X	
Optimization with a fitness function	X		X
Oriented to novice composers	X		X
BDI architecture	X		
Interactivity	X	X	X

To support the contribution of this new expert system, the table below compares our system and two other creative systems: Kirke and Miranda (2015) labeled as System 1 and Navarro et al. (2015) labeled as System 2, briefly explained in Section 2. System 1 uses emotions and MAS to generate a musical melody. The use of emotions to create new music is an original aspect of this system; however, although it interacts with the users, it is not a proper application to assist novice composers in classical music. Furthermore, their scalability and flexibility is left apart in their work, as they do not design the system following a VO methodology.

The second system also provides a tool to assist novice composers by creating harmony, in this case through the use of an evolutionary algorithm. Thus, it does not provide the advantages to work with a MAS explained above. Additionally, the Artificial Immunological System (AIS) used is quite slow compared with our Harmony Search algorithm. A qualitative comparison between the three systems and their features is provided in Table 3.

The listening experiment carried out with the experts and non-experts in music shows that the system generated pleasing sounds within the tonal parameters established as a requirement for novice composers. Additionally, almost all the novice composers questioned in the study stated that the system will be useful for the first steps required to create their own tonal melodies. Thus, this expert system is valuable in terms of creating tonal harmony.

The VO-based design provides a flexibility to this work that can be easily extended, and the BDI architecture permits the use of this system to compose another kind of harmony or even to add a learning component based on the community opinion.

However, some aspects must be improved. The system only creates harmony with a fixed rhythm. It would be desirable to add a rhythmical component in order to be more helpful for the students, as they expressed in the test. The rhythm can give some ideas about some melody changes and directions in the musical context (i.e. if we put a rapid rhythm, the melody would tend to be happier than if the rhythm is slower.) We solved that limitation by adding a regular rhythm as an input that the user has to provide. The present work only creates three-note chords for piano instrumentation, thus it does not consider any other instrument to play the harmony. This can be another point to improve, as specific instrumentation can be important in a more advanced stage of composition learning. Additionally, the system could be incorporated with a melody assistant to be more helpful for the novice composer. However, this falls outside of the scope of the present work.

4. Conclusions and future work

With a general framework of a MAS structure such as PANGEA, we have built a model that can compose different harmonies in order to help students that are new to the field of composition. The use of virtual organizations provided the ability to identify and organize roles, which helped improve the management and thus the efficiency of the composition process. The MAS structure allows an extensible and scalable system to be made as rules, constraints and behavior changed with little effort, and as we continue to search for new ways of mixing different techniques, or even tools in the composition process. The BDI architecture was perfectly suited for the solution we were seeking. BDI has a clear methodology that facilitates the development stage, with many theories that fit our problem. This architecture permits the easily introduction of a learning mechanism, as we can see in our case study. Additionally, the use of PANGEA as a platform allowed for fluid communication between agents, which is evident in the design of the application, and also improved the modularity and the separation between client and provider.

The first experiment carried out demonstrates that our system can compose harmony pleasing (meaning consonant) for a listener and in accordance with the classical rules, which were our constraints. The second experiment highlights that a MAS can develop an expert architecture to create a harmony composition that is useful for a specific community of musicians, in this case, novice composers, which contributes not only to the community of novice musicians, but also to the field of expert systems in music creation. A final contribution is related to the VO applications, meaning that a VO design, which has not been used before in music creation, can be successfully applied to obtain a fully efficient and flexible system in the field of musical generation

Some previous works in Expert Systems can be remarkable. Herremans et al. (2015) create an structured music for bagana, and Ethiopian instrument, based on Markov Models. This proposal can be only applied to rhythm or melodic patrons in baganas. Herremans and Sörensen (2013) create a counterpoint music with a search algorithm. The harmony is not addressed in this paper, and it is not oriented to assist an external user, unlike the system proposed here. Delgado et al. (2009) built Inmamusys, a music system based on agents and expert systems, which is quite similar to the work developed here, but they are interested in music expressing emotions, and not in assist an external user to improve their skills. Additionally, the harmony proposal of this work is based on a small set of rules, which significantly limits the chord construction. Finally, López-Ortega and López-Popa (2012) present an expert system to assist composing musical pieces. This work follows fractal rules, thus it is not able to compose tonal harmony. Additionally, it is not oriented to students that wants to learn tonal music composition.

As a future work, we would like to incorporate rhythms, similar to what was previously presented by Eigenfeldt and Pasquier (2013), based on Markov Models. Another improvement could be the application of the deployment of chords, adapting the rhythms of a composition in different voices. This adaptation might be done with another multiagent system that studies the consonance between each voice in each moment based on a general Markov Model. In the same way, this model can evolve to learn and self-check its own mistakes in harmony composition. We can incorporate the rates given by the experts that evaluate the results of our system, as a feedback that modifies the Beliefs of the BDI architecture. A sound transcription system could also be used to check melody errors by studying the classical rules for the restrictions of consonance or voice leading, thus allowing the system to be more interactive. We will also study the difficulty of the created compositions to adapt the system to the learner, offering students a new way to improve their abilities. This could be done by adding new intentions used in the BDI architecture that change the complexity of the harmony as learners progress in their composition skills.

References

Boden, M. (1987). *Artificial intelligence and natural man*. Cambridge: MIT Press.

Boissier, O., & Gâteau, B. (2007a). Normative multi-agent organizations: Modeling, support and control, draft version. In *Proceedings of seminar 07122 – normative multi-agent systems*.

Boissier, O., Hübner, J. F., & Sichman, J. S. (2007b). Organization oriented programming: From closed to open organizations. In *Proceedings of VII international conference on engineering societies in the agents world* (pp. 86–105). Springer.

Bown, O. (2011). Experiments in modular design for the creative composition of live algorithms. *Computer Music Journal, 35*(3), 73–85.

Cohen, H. (1995). The further exploits of aaron, painter. *Stanford Humanities Review, 4*(2), 141–158.

Colton, S., & Wiggins, G. A. (2012). Computational creativity: The final frontier? In *Proceedings of European conference on artificial intelligence, ECAI* (pp. 21–26).

Cope, D. (1987). An expert system for computer-assisted composition. *Computer Music Journal, 11*(4), 30–46.

Corchado, J. M., Pavón, J., Corchado, E. S., & Castillo, L. F. (2004). Development of CBR-BDI agents: A tourist guide application. In *Advances in case-based reasoning* (pp. 547–559). Springer.

Csikszentmihalyi, M. (1997). *Flow and the psychology of discovery and invention*. New York: Harper Perennial.

Deguet, J., Demazeau, Y., & Magnin, L. (2006). Elements about the emergence issue: A survey of emergence definitions. *ComPlexUs, 3*(1–3), 24–31.

Delgado, M., Fajardo, W., & Molina-Solana, M. (2009). Inmamusys: Intelligent multiagent music system. *Expert Systems with Applications, 36*(3), 4574–4580.

Dignum, M. (2004). A model for organizational interaction: based on agents, founded in logic. Utrecht University Repository.

Dignum, V., Meyer, J.-J., Weigand, H., & Dignum, F. (2002). An organization-oriented model for agent societies. In *Proceedings of international workshop on regulated agent-based social systems: Theories and applications*.

Dubnov, S., & Assayag, G. (2013). Memex and composer duets: computer-aided composition using style mixing. *Open Music composers book*: Vol. 2 IRCAM Ed.

Ebcioglu, K. (1988). An expert system for harmonizing four-part chorales. *Computer Music Journal, 12*, 43–51.

Eigenfeldt, A., Bown, O., & Carey, B. (2015). Collaborative composition with creative systems: Reflections on the first musebot ensemble. In *Proceedings of the sixth international conference on computational creativity, June 2015* (p. 134).

Eigenfeldt, A., & Pasquier, P. (2013). Considering vertical and horizontal context in corpus-based generative electronic dance music. In *Proceedings of the fourth international conference on computational creativity* (p. 72).

Ferber, J., & Gutknecht, O. (1998). A meta-model for the analysis and design of organizations in multi-agent systems. In *Proceedings of international conference on multi agent systems* (pp. 128–135). IEEE.

Ferber, J., Gutknecht, O., & Michel, F. (2004). From agents to organizations: An organizational view of multi-agent systems. In *Proceedings of the IV international conference on agent-oriented software engineering* (pp. 214–230). Springer.

Ferber, R. (2003). *Information retrieval: Suchmodelle und data-mining-verfahren für textsammlungen und das web* dpunkt Verlag, Auflage: 1.

Galland, S., Gaud, N., Rodriguez, S., & Hilaire, V. (2010). Janus: Another yet general-purpose multiagent platform. In *Proceedings of the seventh AOSE technical forum, Paris*.

Geem, Z. W., & Choi, J.-Y. (2007). Music composition using harmony search algorithm. In *Applications of evolutionary computing* (pp. 593–600). Berlin, Heidelberg: Springer.

Geem, Z. W., Kim, J. H., & Loganathan, G. V. (2001). A new heuristic optimization algorithm: harmony search. *Simulations, 76*(2), 60–68.

Giret, A., Julián, V., Rebollo, M., Argente, E., Carrascosa, C., & Botti, V. (2010). An open architecture for service-oriented virtual organizations. In *Programming multi-agent systems* (pp. 118–132). Springer.

Herremans, D., & Sörensen, K. (2013). Composing fifth species counterpoint music with a variable neighborhood search algorithm. *Expert Systems with Applications, 40*(16), 6427–6437.

Herremans, D., Weisser, S., Sörensen, K., & Conklin, D. (2015). Generating structured music for bagana using quality metrics based on markov models. *Expert Systems with Applications, 42*(21), 7424–7435.

Holtzman, S. (1981). Using generative grammars for music composition. *Computer Music Journal, 5*(1), 51–64.

Hoover, A. K., Szerlip, P. A., & Stanley, K. O. (2011). Interactively evolving harmonies through functional scaffolding. In *Proceedings of the 13th annual conference on genetic and evolutionary computation* (pp. 387–394). ACM.

Howden, N., Rönnquist, R., Hodgson, A., & Lucas, A. (2001). Jack intelligent agents–summary of an agent infrastructure. In *Proceedings of the 5th international conference on autonomous agents*.

Hübner, J. (2007). J-Moise+ programming organisational agents with Moise+ & Jason. In *Technical Fora Group at EUMAS: Vol. 7*.

Hübner, J. F., Boissier, O., Kitio, R., & Ricci, A. (2010). Instrumenting multi-agent organisations with organisational artifacts and agents. *Autonomous Agents and Multi-Agent Systems, 20*(3), 369–400.

Josuttis, N. M. (2007). *SOA in practice: The art of distributed system design*. O'Reilly Media, Inc.

Kirke, A., & Miranda, E. (2015). A multi-agent emotional society whose melodies represent its emergent social hierarchy and are generated by agent communications. *Journal of Artificial Societies and Social Simulation, 18*(2), 16.
Lacomme, L., Demazeau, Y., & Dugdale, J. (2010). CLIC: an agent-based interactive and autonomous piece of art. In *Advances in practical applications of agents and multiagent systems* (pp. 25–34). Springer.
López-Ortega, O., & López-Popa, S. I. (2012). Fractals, fuzzy logic and expert systems to assist in the construction of musical pieces. *Expert Systems with Applications, 39*(15), 11911–11923.
Lösch, U., Dugdale, J., & Demazeau, Y. (2009). Requirements for supporting individual human creativity in the design domain. In *Proceedings of the 5th international conference on entertainment computing, ICEC 2009* (pp. 210–215). Springer.
Macedo, L., & Cardoso, A. (2001). Sc-eune-surprise/curiosity-based exploration of uncertain and unknown environments. In *Proceedings of the AISB'01 symposium on emotion, cognition and affective computing* (pp. 73–81). Citeseer.
Macedo, L., & Cardoso, A. (2004). Exploration of unknown environments with motivational agents. In *Proceedings of the third international joint conference on autonomous agents and multiagent systems-volume 1* (pp. 328–335). IEEE Computer Society.
Machado, P., Romero, J., Santos, M. L., Cardoso, A., & Manaris, B. (2004). Adaptive critics for evolutionary artists. In *Applications of evolutionary computing* (pp. 437–446). Springer.
Maher, M. L. (2012). Computational and collective creativity: Who's being creative? In *Proceedings of the 3rd international conference on computational creativity* (pp. 67–71). Citeseer.
Marsden, A. (2000). *Music, intelligence and artificiality*. Amsterdam: Harwood.
Martin, A., Jin, C. T., & Bown, O. (2011). A toolkit for designing interactive musical agents. In *Proceedings of the 23rd Australian computer-human interaction conference* (pp. 194–197). ACM.
McVicar, M., Fukayama, S., & Goto, M. (2014). Autorhythmguitar: Computer-aided composition for rhythm guitar in the tab space. In *Proceedings of the ICMC–SMC joint conference*. In press
Mocholi, J. A., Martinez, V., Jaen, J., & Catala, A. (2012). A multicriteria ant colony algorithm for generating music playlists. *Expert Systems with Applications, 39*(3), 2270–2278.
Monteith, K., Martinez, T., & Ventura, D. (2012). Automatic generation of melodic accompaniments for lyrics. In *Proceedings of the international conference on computational creativity* (pp. 87–94).
Morreale, F., De Angeli, A., Masu, R., Rota, P., & Conci, N. (2014). Collaborative creativity: The music room. *Personal and Ubiquitous Computing, 18*(5), 1187–1199.
Mostafa, M. M., & Billor, N. (2009). Recognition of western style musical genres using machine learning techniques. *Expert Systems with Applications, 36*(8), 11378–11389.
Navarro, M., Caetano, M., Bernardes, G., de Castro, L. N., & Corchado, J. M. (2015). Automatic generation of chord progressions with an artificial immune system. In *Evolutionary and biologically inspired music, sound, art and design* (pp. 175–186). Springer.
Pachet, F. (2002). Playing with virtual musicians: The continuator in practice. *IEEE MultiMedia, 9*(3), 77–82.
Pachet, F. (2003). The continuator: Musical interaction with style. *Journal of New Music Research, 32*(3), 333–341.
Pereira, F. C., Machado, P., & Cardoso, A. (1998). Darwinci: Creating bridges to creativity. In *Advances in artificial intelligence* (pp. 239–248). Springer.
Roads, C., & Wieneke, P. (1979). Grammars as representations for music. *Computer Music Journal, 3*(1), 48–55.
Velardo, V., & Vallati, M. (2014). Automatic melody composition and evolution: a cognitive-based approach. In *Proceedings of conference on interdisciplinary musicology, CIM*.
Zato, C., Villarrubia, G., Sánchez, A., Barri, I., Rubión, E., Fernández, A., et al. (2012). Pangea – Platform for automatic construction of organizations of intelligent agents. In *Distributed computing and artificial intelligence* (pp. 229–239). Springer.