Analysis of Computational Intelligent Techniques of Music Generation

Nermin Naguib J. Siphocly, El-Sayed M. El-Horbaty, Abdel-Badeeh M. Salem

Computer Science Department, Faculty of Computer & Information Sciences
Ain Shams University, Cairo, Egypt.
nermine.naguib@gmail.com, sayed.horbaty@yahoo.com, absalem@cis.asu.edu.eg

Abstract

Computer music generation systems aid musicians in creating and producing high quality music using computers. In the few past years, various intelligent techniques have been used to teach computers ways of generating music either through composition, improvisation, or expressive music playing. This paper presents a comprehensive analysis of the recent publications in music generation field. We particularly focus on music composition applications, especially those adopting algorithmic composition techniques. Our analysis covers the most recently adopted techniques such as deep neural networks and generative adversarial networks. The importance of our analysis is to give insight to researchers of this area about the most suitable techniques to choose from to perform each musical task. Our study shows that the most successful commonly used technique for improvisation is factor oracles. As for music expressiveness which is a rather new field, mathematical models and neural networks achieved the best results. Composition systems have various branches; however, the state-of-the-art technique for composition applications is the generative adversarial networks.

Keywords: *Computer Music, Machine Learning, Computational Intelligence, Algorithmic Composition, Artificial Intelligence*

1. Introduction

Since the invention of computers, utilizing their computational power and precision in the field of music generation and composition has been very tempting. A recent research by B. Jack Copland et al. [1] confirms that the first computer generated musical notes were recorded in the late 40's by Alan Turing; the father of modern computer science himself. The authors' finding defies the myth that the first computer musical notes were heard in the late 50's (Illiac Suite) [2], overlooking the pioneering work of Turing in turning computer into a musical instrument as they described it. Turing discovered that repeating the emission of clicks from the loudspeaker of his Manchester computer with certain patterns, is interpreted by human ears as a continuous sound rather than discrete clicks. Furthermore, he discovered that the different patterns sound like different musical notes such as C_5 and F_4. Later, in 1951, Christopher Strachey, a talented programmer, succeeded in developing a program that generates Britain's national anthem "God Save the King" [3] along with other melodies, which were then recorded by the BBC [4].

As computer prices continued to decrease, many music composers replaced instruments with sets of small-sized computers in their studios. Consequently, in the past few years, research on computer music evolved significantly and became challenging for the Artificial Intelligence (AI) community in particular. AI was heavily involved because teaching computers to create (or assist in the creation of) music needs high levels of creativity.

According to Ramon López [5], current music generation applications can be classified into three categories; *improvisation*, *composition*, and *expressiveness*. Improvisation mentioned in [5], or *interaction* as named by Herremans et al. [6], represents the process where computers generate music in real-time in response to and in harmony with human musicians. The complexity of computer improvisation comes from the fact that there is no time to tune the output music; no preparation nor rehearsals. Composition is the musical task of formulating a new musical piece with all its components which are mainly: melody, accompaniment, and rhythm. Each of the music composition tasks can be performed either fully or partially by a computer. Expressive performance simulation is the study of introducing music players' personal touch to the computer-generated music compositions through some techniques such as phrasing, dynamics, articulation, etc.

In this survey, we try to give a broad idea of the various AI applications, challenges, and research directions within the field of computer music generation. For each music generation category, we list the state-of-the-art systems and intelligent techniques used. We also put forth an objective analysis of these techniques accompanied by the corresponding musical tasks that can be accomplished by applying them. In our analysis, we stress on the strengths and weaknesses of each technique with respect to the musical tasks associated to it. Our work aims to expound the most proper techniques to choose from for each musical task, justifying these choices. We aspire that our work would give insight about future research directions related to the field of computer music generation without ignoring the potential enhancements to the currently existing applications and techniques.

This paper is organized as follows; the next section gives a graphical classification of computer music generation applications without digging into technical details. Sections 3, 4, and 5 elaborate on the applications and AI techniques adopted in the three main music generation categories; improvisation, composition, and expressiveness respectively. At the end of each of these sections, an analysis of the implemented techniques is included highlighting the pros and cons of each. We discuss results of our analysis in Section 6 and conclude our study in Section7 recounting some thoughts about what would come next in the field. In the future we intend to prepare a detailed survey for the algorithms used in computer music composition. On the other hand, we intend to implement and compare between groups of algorithms in the field as possible.

2. Classification of Computer Music Applications

In 2006, Ramon López [5] classified the applications of computer music into three major categories; music *composition*, synthesizing *expressive*musical performances, and music *improvisation*. To our knowledge, most of computer music applications lie under the umbrella of these three categories. Our survey discusses the definition and the advancements in each of the three categories while going in depth into the category of music composition. Proceeding from López's categorization, the diagram in Figure1 represents a classification for the different computer music generation systems in general and for the music composition applications in specific.

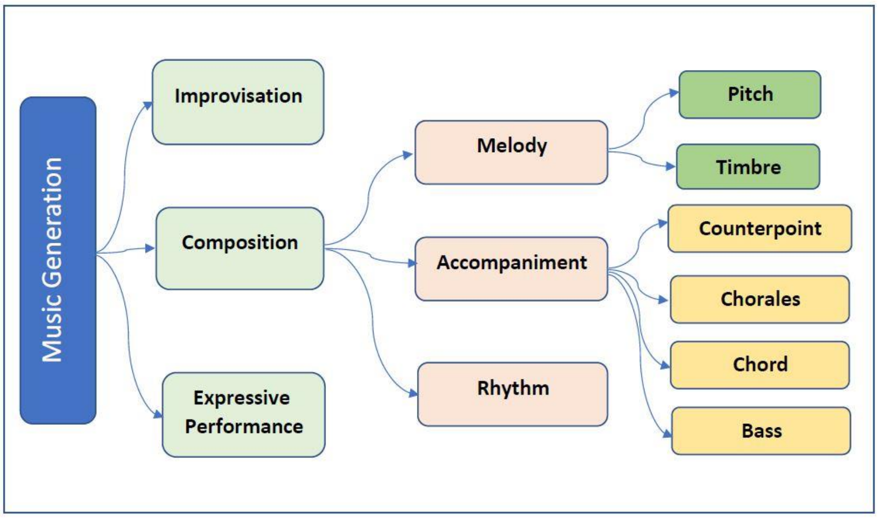

Figure 1. **Classification of Computer Music Generation Systems**

The diagram from its left side divides *music generation* systems into three categories; either *improvisation* systems which mainly provides interaction between computer musician and a human musician (discussed in Section3), or music *composition* systems that are responsible for generating musical pieces of different features and tastes (discussed in Section4), or the last category; the *expressive performance* systems which simulates the personal touch of human musicians by computers (discussed in Section5).

Music composition systems deals with automating the different components of a musical piece, which mainly are:

- The *melody* which is the main musical notes of the piece,
- The *accompaniment* or the notes that accompanies the main melody and are in harmony with it, and the *rhythm* or the beats of the musical piece.

The melody is represented by notes *pitch* and *timbre*. On the other hand, there are various types of accompaniment music such as: *counterpoint*, *chorale* harmonization, *chord*, and the *bass* accompaniment which will be all discussed in Sections 4.1 to 4.7.As follows we outline the modern and pioneering research in each of the music generation categories (improvisation, composition and expressiveness) analyzing the used AI techniques; their strengths and weaknesses.

3. Study of Music Improvisation Systems

A pioneering work in music improvisation includes "The Continuator", an improvisation system developed by Francois Pachet [7], using Markov model of musical styles. Later, Gil Weinberg and Scott Driscol [8] merged improvisation with robotics by creating a robotic drum-player using a bunch of already existing rhythm analysis and generation algorithms in their work. In the following years, Weinberg along with other researchers extended the work on robotic musicians and developed a robotic Xylophone player utilizing genetic algorithms (GAs) [9].

Concurrently, OMAX-OFON [10] was developed that uses statistical learning method to create an improvisation musician-machine. The architecture of OMAX-OFON is hybrid between two famous musical frameworks; Max and Open Music communicating together.

OMAX is based upon the Factor Oracle (FO) model which is a finite state automaton originally developed for effectively searching for substrings (factors) in a given text. G. Assayag et al. did further research with FO in [11,12]. Arshia Cont et al. [13] developed an anticipatory machine improvisation system applying reinforcement learning aiming to mimic a human musician style. More recently, Nika et al. [14] developed Import K; a guided human-computer improvisation system which has a reactive architecture, also in terms of FO. Subsequently, Ken Déguern el et al. [15] developed two methods for improvisation systems. The first system combines interpolated probabilistic models with a FO. The second is a multidimensional system that is built on probabilistic message passing between dimensions (a cluster graph).

Table 1. Summary of AI techniques used in Improvisation

TECHNIQUE	REFERENCES	STRENGTHS	WEAKNESSES
Misc.*	2006[8]	Used in rhythm interaction, used in robotic musicianship**.**	A percent of noise in the generated rhythm**.**
Genetic Algorithm	2008[9]	Used in robotic musicianship, enables robot to make decisions about playback modes**.**	Needs optimization to decreasethe response time**.**
Unsupervised Learning	2007[13]	Reach an anticipation without prior knowledge**.**	Might need a lot of training.
Markov Chains	2003[7]	Imitate musical style, continue on a given musical input**.**	Time consuming**.**
Factor Oracle	2006 [10],2007 [13], 2010 [11], 2012[12], 2015 [14],2018 [15]	Achieves certain formal planning with only small data available, can combine between reactivity and anticipation, can learn the correlation between musical dimensions**.**	Produced music is not rich**.**

**The author used a collection of already existing machine learning algorithms*

Table 1 summarizes the AI techniques used in developing computer music improvisation systems since the year 2003. Researches applying more than one technique appears more than once in the table. Experimenting with miscellaneous already existing machine learning algorithms in [8] resulted in robotic musicianship and rhythm interaction with fair results having a bit noisy nature.

Genetic algorithms are used in robotics musicianship and allows for robot's decision making. For example, [9] fitness function enables the robot to choose which melody to play next. The problem with using GAs for real-time improvisations is that in order to minimize the response time, the generated melodies should be short in length and the number of candidates should be limited. Unsupervised learning helps in reaching a musical anticipation by the computer through training on small amounts of data without any prior musical knowledge [13]; however, it might need a lot of training to reach that anticipation. Markov Model help in imitating a music player's style or continue on his/her input playing. As an example, Pachet's system [7] was developed to be able to either imitate a certain musical style, perform as a continuation of a musician's input, or as interactive improvisation back up. Markov chains are time consuming so this might affect the response time, that's why they are more suitable for style imitation or music continuation not real-time response playing.

Factor Oracle has a big share of improvisation applications and proves to be suitable for the task. It has the advantage of achieving certain formal planning (anticipation) with only small data available through unsupervised learning and no prior information of formal structure for any musical style [13]. FO also allows for combining both reactivity and anticipation in the music generation process. For example, [14] works by embedding offline processes into a reactive framework; it is controlled by a scenario so that it generates anticipations ahead of time before performance. The generated music is rewritten or refined over time in reaction to external controls or parameters or scenario alteration. In [15], FO were applied to learn the correlation between dimensions (which represent musical features such as melody, harmony, etc.) by training probabilistic models on a corpus of musical work , and these correlations were afterwards used to guide the search in the FO to ensure a logical improvisation. The improvisations generated by FO is still not rich enough to satisfy human ears.

4. Analysis of Music Composition Systems

Computer music composition became highly correlated with the term "algorithmic composition" which, from its name, uses algorithms in automating the music composition process. In 2013, Jose and Francisco [16] gave a comprehensive survey on algorithmic composition where they listed, giving examples, the various techniques used in thatfield. The authors concluded their survey that each technique alone was not enough for generating satisfying compositions and that it is rather worth studying the combination of various techniques. The following subsections discuss recent research in each composition task.

4.1 Melody Pitch

When composing a musical piece, one of the first elements a musician would think about is the main melody of that piece. The melody is a set of single consecutive, satisfying, notes forming the musical piece. Musical notes are mainly distinguished by their pitch which is the human interpretation to sound wave frequency and is closely related to it. Many music generation systems aim to imitate human composers' experience; thus Case-Based Reasoning (CBR) makes a suitable approach for the task. María Navarro-Cáceres et al. [17]developed a case-based reasoning architecture along with a Markov model to obtain the probabilities of a given note following the last note incorporated in the melody. The system gets a user feedback through a mechanical device connected to it to control the pitches and the duration of the musical notes.

In their survey, Jose et al. survey [16] hinted about the need for multi-objective fitness function for evolutionary algorithms corresponding to the multiple factors affecting music generation. Pedro et al. [18] proposed a multi-objective fitness function for enhancing the selection process of melody generation. Moreover, they proposed "Melodic Trees" as a data structure for chromosomes representation. Although explored the use of Artificial Neural Networks (ANNs) in music composition, Jose's survey [16] did not mention much about Deep Neural Networks and Deep Learning which has been extensively used in research since then. Florian et al. [19] developed a system for melody generation through deep recurrent neural networks (RNNs) that is able to capture long-range temporal structure. Benjamin Smith [20] took advantage of the conditional restricted Boltzmann machine (CRBM)ability to model temporal dependencies to achieve full reconstructions of musical pieces given a few starting seed note. Recently, Generative Adversarial Networks (GANs) lead to a major breakthrough in many fields, including music generation. Wave Net [21] which is a deep neural network for generating raw audio waveforms is the first application of GANs dealing with audio data. Wave Net was primarily developed for speech synthesis purposes; however, when experimented on musical piano pieces it was able to produce realistic musical waveforms. Li-Chia Yang [22] also utilized GANs for melody generation but worked on MIDI sequences in the symbolic musical space. Hao-Wen Dong et al. [23] developed a GAN for multi-tack music generation. Gong Chen et al. [24] is another example for utilizing GANs in algorithmic composition.

4.2 Melody Timbre

Another aspect of musical piece melody is "Timbre" which is more related to musical instruments sounds interpretation. The most common definition of timbre is that it is the quality or tone of sound and it distinguishes different voices; thus, so important in our interpretation of musical sounds and instruments. Daniel Mintz [25] developed a method for timbral synthesis based on timbral description scheme that is more standard than the verbal description suggested earlier, which is MPEG-7. In Daniel's system, users can specify the timbre they want using standardized descriptors. The timbral synthesis engine converts these descriptor values into control envelopes based on linear optimization in order to transform analysis equations into synthesis equations. The coefficients of these equations allows for precise specification of points in a timbral space. Further working with timbral spaces, Allan Seago et al. [26] proposed a timbre space search strategy, based on weighted centroid localization (WCL). Seago expanded on his work in [27].

Timbre's effect is shown clear in the process of orchestration in music composition; which is even more complicated than sound synthesis because in this case each sound's timbre contributes in the overall interpretation of the generated music. Carpentier et al. [28] introduced Orchidée, which is a time-efficient evolutionary orchestration algorithm that allows the discovery of optimal solutions and favors the exploration of non-intuitive sound mixtures. In a very recent research, Marcelo Caetano et al. [29] developed a computer aided orchestration system using an artificial immune system (AIS) called Opt-aiNet. Opt-aiNet is formulated as a multi-modal system that searches for combinations of musical instrument sounds that minimize the distance to a reference sound encoded in a fitness function.

4.3 Counterpoint Accompaniment

Counterpoint is a special type of harmony or music accompaniment that has a set of strict rules to generate multiple accompanying voices (typically two to four). Thus, rule-based systems were convenient for this type of harmony. Gabriel Aguilera et al. [30] used Derive 6, a computer algebra system, to code counterpoint rules by means of probabilistic logic.

Away from rule-based systems, Victor Padilla et al. [31] recently developed an imitative system for generating two-voice counterpoint based on Palestrina-style. Their system combines statistical generation and pattern discovery. Once patterns are organized according to a probabilistic distribution, two-voice counterpoint is generated into those patterns using a first-order Markov model. Subsequently, Cheng et al. [32] intended by their work to imitate the way of real composers in writing music in a nonlinear fashion rather than chronological one adopted by the earlier systems. They trained a convolutional neural network to generate partial musical scores, moreover, they introduced the use of blocked Gibbs sampling as an analogue to rewriting.

4.4 Chorale Accompaniment

Chorale accompaniment is the type of harmonization that is formed of four-part music lines; soprano and three other lower voices. Recent research by F. Liang et al. [33] presents BachBot; a deep LSTM generative model for producing music in the style of Bach's chorales. G. Hadjeres et al. [34] also developed a system that imitates Bach's chorales; however, they used a dependency network and pseudo-Gibbs for the music sampling. T. Yamada et al. [35] compared between the use of Bayesian networks (BNs) and recurrent neural networks in chorale music generation highlighting the strengths and weaknesses of each.

4.5 Chord Accompaniment

Chord accompaniment is the most leading type of harmony. It is the set of multiple harmonic notes that sound simultaneous when heard. Several techniques were used in chord accompaniment generation; Liu et al. [36] developed a genetic algorithm (GA) to generate polyphonic accompaniment, in which the fitness function consists of several evaluation rules based on music theory instead of human feedback. They later [37] introduced the use of data mining to extract chord patterns and styles from specific composer's music. The extracted patterns were fed as genes into the GA whose fitness function is also based on music theory. Concurrently, Navarro et al. [38], also embedded music theory in their system through developing a penalty function that encodes musical rules to automatically generate chord progressions. They also made use of artificial AIS in order to propose candidates for the next chord in a sequence that minimize the penalty function. The AIS enabled for proposing several candidates due to finding multiple optima in parallel.

Neural Networks also have a good share of recent chord generation applications; Brunner et al. [39] recently developed an approach for polyphonic music generation that is based on LSTMs. Music is generated in twosteps; first, a chord LSTM predicts a chord progression based on a chord embedding. A second LSTM then generates polyphonic music from the predicted chord progression. A very recent research by M. Nadeem [40] deploys LSTM Recurrent Neural Networks to generate musical notes along with their chords at the same time using a fixed time-step with a view to improve the quality of the music generated. Separate layers of chord and note data are trained in parallel before combining the output of those layers (through a dense layer followed by a final LSTM layer) to produce new music. This technique ensures that both inputs, notes and chords, are considered at all steps of generation, and thus are closely related.

4.6 Bass Accompaniment

Generating Bassline is another aspect of accompaniment music that is closely related to chord accompaniment. In addition to main and chord accompaniment generation, Liu et al. [36] used GA to generate Bass line. Their system conducted GA three times with different fitness functions for each of the polyphonic accompaniments aspects. K. Komatasu et al. [41] enhanced on their work in [42] by adding a bass-line generation model through genetic programming.

4.7 Rhythm

Rhythm represents the beats of the musical piece it determines the speed and style of the piece. Since Rhythmis a regularly repeated pattern of musical beats, recent research of rhythm composition is closely related to pattern mining. Liu et al. [37] developed an evolutionary composition system based on genetic algorithms and pattern mining. The extracted patterns then serve as the basis for chromosome representation. Long et al. [43] used pattern mining in order to develop a melody composer (T-Music) that utilizes the correlation between a melody and its lyric to compose new music given a lyric. Finding lyric-note correlations is based on performing a frequent pattern mining on a large database containing existing songs each of which involves both its melody and its lyric. Earlier efforts of style imitation include Chiu et al. [44] music composition approach that utilizes pattern mining techniques to discover the rules of music composition characterized by the music properties, music structure, melody style and motif.

Table 2. Summary of AI techniques in music composition

TECHNIQUE	APPLIED IN	STRENGTHS	WEAKNESSES
Genetic Algorithms	*Pitch*: 2016 [18] - *Timbre*: 2010 [28] - *Chord*: 2012 [36], 2015 [37] -*Bass*: 2012 [36], 2015 [41] – *Rhythm*: 2015 [37]	Close to natural composition process, can be conducted to perform multiple compositional tasks in parallel.	Single fitness function is not enough to select the best musical solution, while multi-objective fitness functions are sometimes contradictory and combining them needs optimization.
Case-Based Reasoning	Pitch: 2017 [17]	A means of capturing human musical experiences.	Not enough alone to generate satisfying results, needs external guidance.
Artificial Neural Networks	*Counterpoint*: 2019 [32] - *Chorale*:2017 [34],2018 [35] - *Chord*:2017 [39], 2019 [40]	Produce almost pleasant music, can produce polyphonic music, RNNs And CRBM are capable of capturing long term dependencies in music, capable of producing four-part harmonization, convolution networks allow for revisiting generated compositions for enhancement.	BN generated unsmooth bass lines, while the RNN generated alto and tenor lines that tend to be monotonous, some network types such as dependency networks are time consuming.
Deep Neural Networks	*Pitch*: 2016 [19], 2017 [20] - *Chorale*: 2017 [33]	Enable training on large musical corpus, can be conducted to perform multiple compositional tasks in parallel, ability to extract musical features from data.	Sometimes the network is not deep enough for getting satisfying results which needs more computation.
Linear Optimization	*Timbre*:2007 [25],2010[26],2013 [27]	More standard than verbal descriptors.	Performs better in lower dimensions timbral spaces.
Artificial Immune System	*Timbre*: 2019[29] - *Chord*:2015 [38]	Gives more than one solution.	Needs threshold function if fewer solutions are needed.
Rule-Based	*Counterpoint*: 2010 [30]- *Chord*: 2015 [38]	Easier to develop due to existence of musical rules, can assist other techniques better results.	Are not enough alone.
Markov Chains	*Pitch*: 2017 [17] - *Counterpoint*: 2018 [31]	Help in predicting the next note.	Does not capture long term musical dependencies, not suitable for polyphonic music, time consuming.
Generative adversarial Networks	*Pitch*: 2016 [21], 2017 [22], 2018 [23], 2018 [24]	Applicable on monophonic and polyphonic music, can be applied to symbolic or waveform music, generated music is highly rated, multiple adversarial networks can be applied.	Needs high computational power, multiple adversarial networks need optimization to be combined.

Table 2 gives a brief summary for the AI techniques used in music composition since the year 2007. Genetic algorithms can be applied in melody pitch generation, timbre orchestration, chord, bass line and rhythm generation. They are close to the natural process of music composition; "*For composers, it provides an innovative and natural means for generating musical ideas from a specifiable set of primitive components and processes. For musicologists, these techniques are used to model the cultural transmission and change of a population's body of musical ideas over time*" [45].GAs also have the advantage of being able to perform several composition tasks in parallel such as melody and chord generation [41], or polyphonic accompaniment [36, 37] (formed of main, bass, and chord). The problem with genetic algorithms for music generation is always in finding the best fitness function. In most of the cases a single fitness function is not enough to judge the multi-facets nature of music. Thus, a multi-objective fitness function is needed for the task, the problem with this approach is that all the fitness evaluations (sometimes contradictory) need to be combined formulating an optimization problem.

Case-based reasoning CBR can be used in melody pitch generation specially in imitation systems as it represents a means of capturing human past composition experiences that can then be used to produce new melodies of the same style. None the less, CBR need external guidance for the generated pitches and duration to get satisfying results [17].

Artificial Neural Networks ANNs along with Deep Neural Networks are applied in various composition tasks such aschorale, counterpoint and chord accompaniment in addition to melody pitch generation. When evaluated by human listeners, these techniques are proved to produce appealing music to the ears to a great extent [32-34, 40]. ANNs are able to produce polyphonic music [39] as well as four-part chorales [34, 35]. Recurrent neural networks and conditional restricted Boltzmann machine (CRBM) are capable of capturing long-term dependencies between notes so as to generate notes consonant with previous ones. Convolution Networks tend to be more realistic in simulating the composition process, not chronologically, but rather in an arbitrary manner revisiting and enhancing previously generated melody parts [32]. In addition to ANNs strengths, deep NNs have the advantage of enabling training on large musical corpus [19]. Moreover, they can perform composition tasks in parallel such as generating rhythm and melody [19]. Deep NNs are successful data-driven models; they are able to extract musical features from the given (training) data only without any prior musical knowledge, then produce compositions of the same style thus providing more generalization [19, 20, 33]. The problem with deep NNs is that sometimes they are not deep enough to produce satisfying results which leads to adding more layers and consequently consuming more computational power [20]. The comparison between RNNs and BNs for chorale generation [35] showed that BN generated unsmooth bass lines although produced consonant harmonies in general. On the contrary the RNN generated smoother bass lines but the alto and tenor lines tended to be monotonous. Hence, this comparison shows how problematic it is to choose which ANN to adopt for the desired musical tasks. Another problem with ANNs is that some network types such as dependency networks are time consuming [34].

Linear Optimization was proved to be better than the early verbal descriptor for timbre synthesis; however, it is limited to small dimensional spaces. In [26] it is stated that WCL method performs significantly better in relatively simple three-dimensional spaces (in this case the formant space and the SCG-EHA spaces) than in spaces where the dimensionality is greater (the MDS space).

Artificial Immune System was adopted in Opt-aiNet [29] for timbre orchestration. Since orchestration can (and actually preferable) to have more than one possibility, Opt-aiNet was developed to maximize diversity in the solution set of multi-modal optimization problems, which results in multiple alternative orchestrations for the same reference sound that are different among themselves. However, in other cases, producing multiple solutions (multiple optima) in parallel might sometimes need some kind of filtration such as the case with chord generation [38] that needed a penalty (threshold) function to decide for the good chords.

Rule-based systems is simply the coding of music theory rules which makes it the easier to develop. Since learning music theory is not enough for humans to compose, rule-based approach is not enough; that's why it is best to be used in assistance with other techniques such as GAs in the fitness function.

Markov chains are suitable for predicting new notes based on the previous ones, but it is limited to only one preceding note and hence does not support long term dependencies. As the musical voices increase, Markov model will grow significantly; thus, it is more suitable for monophonic rather than polyphonic music composition.

Generative adversarial networks are the state-of-the-art in music composition. Already existing experiments shows that the music generated through GANs are highly ranked by human listeners. GAN shave the advantage of being able to generate monophonic or polyphonic music such as [24] in addition to working with symbolic [22-24] and waveform [21]. The ability of having multiple adversarial networks enable enhancing the generated music based on multiple features and factors but combining these network needs optimization. The problem with GANS is the need for high computational power.

5. Study of Music Expressiveness Systems

Case Based Reasoning (CBR) had a great share of the early research related to expressive performance simulation where the system learns from human players experience. For example; Arcos et al. [46-49] works through the years 1998-2004, in addition to Suzuki et al. work [50].

Some researchers formulated musical expressiveness mathematically such as Carlos and Cancino [51] who proposed an expressive automatic accompaniment system based on basis function models (BMs). As defined by the authors, basis functions are "*numeric features that capture specific aspects of a musical note and its surroundings*". Structural properties of a musical piece (given as a score), which are claimed to affect performance decisions, can be modeled simply via basis functions. Similarly, M. Della Ventura [52] relied on mathematics to develop an algorithm that can investigate the musical expressiveness of a musical piece by reading the score on its symbolic level. Instead of manually modeling the expressiveness, the algorithm identifies the harmonic functions and consequently provides indications for the dynamics by means of graphic representation. Harmonic functions are identified based on a musical grammar which is reflected in the functional harmony.

Sergio Giraldo and Rafael Ramírez [53] used fully connected ANNs to be trained to produce expressive (ornamented) Jaz performance from non-expressive score. The input is the score and its relevant performance by professional guitarist. They set a comparison between different machine learning techniques for the same mentioned task.

Table 3. Summary of AI techniques used in expressiveness

TECHNIQUE	REFEREN-CES	STRENGTHS	WEAKNESSES
Case-Based Reasoning	2004 [49]	Perfect for learning instrument playing by example.	Needs more musical information.
Basis Functions	2001 [51] - 2018 [52]	Contribute in building a predictive model of performance.	Needs sampling technique to be time efficient.
Artificial Neural Networks	2016 [53]	Imitate musical style (Jaz).	Music genre specific.

Table 3 summarizes the AI techniques used in expressive performance since the year 2004. CBR is very suitable for this task because they resemble humans' way of learning by example; however, example only is not enough so we suggest adding musical knowledge to assist in the simulation process. As for BFM, the generated basis functions contribute in building a predictive model of performance that can predict appropriate patterns for expressive performance dimensions such as tempo, timing, dynamics, and articulation. BFM needs sampling, such as Gibbs sampling, to achieve the desired results. Finally, ANNs can be trained to imitate a musical expressiveness style; nonetheless, the network cannot be generalized to every music genre.

6. Results and Discussion

This section discusses the results interpreted from our analysis of the recent publications in music generation field. Music improvisation applications focus on the use of GAs, unsupervised learning, Markov chains, and FOs techniques. Factor Oracles has the greatest share of improvisation applications; however, the resulting improvisations are not rich enough to satisfy human ears. Enhancements might be through combining other techniques with the FO such as following it with an ANN that evaluates the richness of the generated improvisations.

As for music expressiveness, it is a rather new field of research that needs more work to cover all the music genres and get closer in simulating human instrument players' expressiveness. Current music expressiveness systems rely either on CBR, on basis functions, or on neural networks techniques. Future research needs to focus on data-driven approaches to achieve better results in music expressiveness.

Finally, music composition tackle the problem of automation of each composition task through adopting various AI techniques among which: GAs for pitch, timbre, bass, chord, and rhythm generation; CBR for pitch generation; ANNs for counterpoint, chorale, and chord accompaniment; Deep NNs for pitch generation and chorale accompaniment; Linear Optimization for timbre generation; AIS for timbre orchestration and chord accompaniment; Rule-base for counterpoint and chord accompaniment; Markov chains for pitch generation and counterpoint accompaniment; and GANs for pitch generation.

Although recent surveys such as [6] and [54] do not mention the advancements of GANs, this intelligent technique represents a promising research path, not only for music composition but for music generation systems in general. GANs provides for automatic evaluation of the generated music from multiple points of view, not to mention the numerous opportunities for experimenting with GANs to perform a diversity of musical tasks. Moreover, already existing experiments shows that the music generated through GANs are highly ranked by human listeners.

7. Conclusion and Future work

Music generation applications span a wide spectrum between improvisation, composition, and expressiveness categories. The main objective of this paper is to propose a comprehensive analysis of the recent research in music generation systems. We study the recent AI techniques used in music generation systems, focusing on music composition applications, highlighting the weaknesses and strengths of each technique. Furthermore, we aspire to shed the light on potential enhancements to the currently existing music generation systems and on promising future research directions related to that field.

Our study shows that the most successful commonly used technique for improvisation is factor oracles although it is preferable to combine it with other AI techniques to generate richer music. As for music expressiveness which is a rather new field, mathematical models and neural networks achieved the best results so far. However, future research in expressiveness needs to focus on data-driven approaches. Composition systems have various branches corresponding each musical task. Nonetheless, the state-of-the-art technique for composition applications is the generative adversarial networks which were proven to generate music most satisfying to human listeners.

Future work includes preparing a detailed survey for the algorithms used in computer music composition. On the other hand, we intend to implement, combine and compare between groups of algorithms in the field as conceivable.

References

[1]. Copeland B.J. and Long J., editors. *Turing and the History of Computer Music*. In: Floyd J., Bokulich A.(eds) Philosophical Explorations of the Legacy of Alan Turing. Boston Studies in the Philosophy and History of Science, vol 324. Springer, Cham, 2017.

[2]. Lejaren Hiller and Leonard Isaacson. *Machine models of music*. Chapter in Musical Composition with a High-speed Digital Computer, pages 9-21. MIT Press, Cambridge, MA, USA, 1992.

[3]. N. Foy. *The word games of the night bird (interview with Christopher Strachey)*. Computing Europe, page 10-11,15 August 1974.

[4]. Dean and R.T. (ed.), editors. *The Oxford Handbook of Computer Music*. New York: Oxford University Press,2009.

[5]. Ramon López de Mántaras. *Making Music with AI: Some examples*, pages 90-100. IOS Press, 2006.

[6]. Dorien Herremans, Ching-Hua Chuan, and Elaine Chew. *A functional taxonomy of music generation systems.*ACM Comput. Surv., 50(5):69:1-69:30, September 2017.

[7]. François Pachet. *The continuator: Musical interaction with style*. Journal of New Music Research, 32(3):333-341, 2003.

[8]. Gil Weinberg and Scott Driscoll. *Toward robotic musicianship*. Compute. Music J., 30(4):28-45, December 2006.

[9]. Gil Weinberg, Mark Godfrey, Alex Rae, and John Rhoads. *Computer music modeling and retrieval*. Sense of sounds. chapter A Real-Time Genetic Algorithm in Human-Robot Musical Improvisation, pages 351-359.Springer-Verlag, Berlin, Heidelberg, 2008.

[10]. G. Assayag, G. Bloch, and M. Chemillier. Omax-ofon. In Proceedings of the Conference on Sound and MusicComputing, 2006.

[11]. Gérard Assayag, George Bloch, Arshia Cont, and Shlomo Dubnov. *Interaction with Machine Improvisation*, pages 219{245. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[12]. Shlomo Dubnov and Gérard Assayag. *Music design with audio oracle using information rate*. In Proceedings ofthe Conference on Musical Metacreation: Papers from the 2012 AIIDE Workshop, 2012.

[13]. Arshia Cont, Shlomo Dubnov, and Gérard Assayag. *A framework for anticipatory machine improvisation and style imitation*. In in Anticipatory Behavior in Adaptive Learning Systems, August 2007.

[14]. Jérôme Nika, Dimitri Bouche, Jean Bresson, Marc Chemillier, and Gérard Assayag. *Guided improvisation as dynamic calls to an offline model*. In Sound and Music Computing (SMC'15), 2015.

[15]. K. Déguernel, E. Vincent, and G. Assayag. *Probabilistic factor oracles for multidimensional machine improvisation*. Computer Music Journal, 42(02):52-66, June 2018.

[16]. Jose David Fernández and Francisco Vico. *AI methods in algorithmic composition: A comprehensive survey*. J. Artif. Int. Res., 48(1):513-582, October 2013.

[17]. María Navarro-Cáceres, Sara Rodríguez, Diego Milla, Belén Pérez-Lancho, and Juan Manuel Corchado. *A user-controlled system for the generation of melodies applying case-based reasoning*. In David W. Aha and Jean Lieber,editors, Case-Based Reasoning Research and Development, pages 242-256, Cham, 2017. Springer International Publishing.

[18]. Pedro J. Ponce de León, José M. Iñesta, Jorge Calvo-Zaragoza, and David Rizo. *Data-based melody generationthrough multi-objective evolutionary computation.* Journal of Mathematics and Music, 10(2):173-192, 2016.

[19]. Florian Colombo, Samuel P. Muscinelli, Alexander Seeholzer, Johanni Brea, and Wulfram Gerstner. Algorithm iccomposition of melodies with deep recurrent neural networks. CoRR, abs/1606.07251, 2016.

[20]. Benjamin D. Smith. *Musical deep learning: Stylistic melodic generation with complexity-based similarity*. In Musical Meta creativity Workshop at the Eighth International Conference on Computational Creativity, Atlanta,GA, 2017.

[21]. Aäron van den Oord, Sander Dieleman, Heiga Zen, Karen Simonyan, Oriol Vinyals, Alex Graves andNal Kalchbrenner, Andrew Senior, and Koray Kavukcuoglu. *WaveNet: A generative model for raw audio*, arXiv preprint arXiv:1704.01279, 2016.

[22]. Li-Chia Yang, Szu-Yu Chou, and Yi-Hsuan Yang. *MIDINET: A convolutional generative adversarial network for symbolic-domain music generation*. In International Society of Music Information Retrieval (ISMIR), Suzhou,China, 2017.

[23]. Hao-Wen Dong, Wen-Yi Hsiao, Li-Chia Yang, and Yi-Hsuan Yang. *MuseGAN: Multi-track sequential generative adversarial networks for symbolic music generation and accompaniment*. 2018.

[24]. Gong Chen, Yan Liu, Sheng-hua Zhong, and Xiang Zhang. *Musicality-Novelty generative adversarial nets for algorithmic composition*. In Proceedings of the 26th ACM International Conference on Multimedia, MM '18,pages 1607-1615, New York, NY, USA, 2018. ACM.

[25]. D. Mintz. Toward Timbral Synthesis: *A New Method for Synthesizing Sound Based on Timbre Description Schemes*. University of California, Santa Barbara, 2007.

[26]. Allan Seago, Simon Holland, and Paul Mulholland. *A novel user interface for musical timbre design*. In Audio Engineering Society Convention 128, May 2010.

[27]. Allan Seago. *A New Interaction Strategy for Musical Timbre Design*, pages 153-169. Springer London, London,2013.

[28]. Grégoire Carpentier, Gérard Assayag, and Emmanuel Saint-James. *Solving the musical orchestration problem using multi objective constrained optimization with a genetic local search approach*. Journal of Heuristics,16(5):681-714, Oct 2010.

[29]. Marcelo Caetano, Asterios Zacharakis, Isabel Barbancho, and Lorenzo J. Tardón. *Leveraging diversity incomputer-aided musical orchestration with an artificial immune system for multi-modal optimization*. Swarmand Evolutionary Computation, 2019.

[30]. Gabriel Aguilera, José Luis Galán, Rafael Madrid, Antonio Manuel Martínez, Yolanda Padilla, and Pedro Rodríguez. *Automated generation of contrapuntal musical compositions using probabilistic logic in derive*. Mathematics and Computers in Simulation, 80(6):1200 - 1211, 2010. Fifth IMACS Seminar on Monte Carlo Methods Applications of Computer Algebra 2007 (ACA 2007) special session on Nonstandard Applications of Computer Algebra Computational Biomechanics and Biology, a collection of papers presented at the 1st IMACS International Conference on the Computational Biomechanics and Biology ICCBB 2007.

[31]. Victor Padilla and Darrell Conklin. *Generation of two-voice imitative counterpoint from statistical models*. International Journal of Interactive Multimedia and Artificial Intelligence (IJIMAI), 5:22-32, 2018.

[32]. Cheng-Zhi Anna Huang, Tim Cooijmans, Adam Roberts, Aaron C. Courville, and Douglas Eck. *Counterpoint by convolution*. CoRR, abs/1903.07227, 2019.

[33]. Feynman T. Liang, Mark Gotham, Matthew Johnson, and Jamie Shotton. *Automatic stylistic composition ofBach chorales with deep LSTM*. In 18th International Society for Music Information Retrieval Conference (ISMIR),2017.

[34]. Gaëtan Hadjeres, François Pachet, and Frank Nielsen. *DeepBach: A steerable model for Bach chorales generation*.In Proceedings of the 34th International Conference on Machine Learning - Volume 70, ICML'17, pages 1362-1371. JMLR.org, 2017.

[35]. Tatsuro Yamada, Tetsuro Kitahara, Hiroaki Arie, and Tetsuya Ogata. *Four-part harmonization: Comparison of a Bayesian network and a recurrent neural network*. In Mitsuko Aramaki, Matthew E. P. Davies, RichardKronland-Martinet, and Sølvi Ystad, editors, Music Technology with Swing, pages 213-225, Cham, 2018. Springer International Publishing.

[36]. Chien-Hung Liu and Chuan-Kang Ting. *Polyphonic accompaniment using genetic algorithm with music theory*.2012 IEEE Congress on Evolutionary Computation, pages 1-7, 2012.

[37]. Chien-Hung Liu and Chuan-Kang Ting. *Music pattern mining for chromosome representation in evolutionary composition*. 2015 IEEE Congress on Evolutionary Computation (CEC), pages 2145-2152, 2015.

[38]. María Navarro-Cáceres, Marcelo Caetano, Gilberto Bernardes, Leandro Nunes de Castro, and Juan Manuel Corchado. *Automatic generation of chord progressions with an artificial immune system*. In Colin Johnson, Adrian Carballal, and Jo~ao Correia, editors, Evolutionary and Biologically Inspired Music, Sound, Art and Design, pages 175-186,Cham, 2015. Springer International Publishing.

[39]. G. Brunner, Y.Wang, R.Wattenhofer, and J. Wiesendanger. *Jambot: Music theory aware chord based generation of polyphonic music with LSTMs*. In 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), pages 519-526, Nov 2017.

[40]. M. Nadeem, A. Tagle, and S. Sitsabesan. *Let's make some music*. In 2019 International Conference on Electronics, Information, and Communication (ICEIC), pages 1-4, Jan 2019.

[41]. Kanae Kunimatsu, Yu Ishikawa, Masami Takata, and Kazuki Joe. *A music composition model with genetic programming -a case study of chord progression and bassline-*. In International Conference on Parallel and Distributed Processing Techniques and Applications, PDPTA'15, pages 256-262, Monte Carlo Resort, Las Vegas,USA, July 27-30 2015.

[42]. Kyoko Komatsu, Tomomi Yamanaka, Masami Takata, and Joe Kazuki. *A music composition model with genetic programming*. In Hamid R. Arabnia, Steve C. Chiu, George A. Gravvanis, Minoru Ito, Kazuki Joe, HiroakiNishikawa, and Ashu M. G. Solo, editors, Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications, (PDPTA 2010), pages 686-692, Las Vegas, Nevada, USA, July 12-152010. CSREA Press.

[43]. Cheng Long, Raymond Chi-Wing Wong, and Raymond Ka Wai Sze. *T-music: A melody composer based on frequent pattern mining*. 2013 IEEE 29th International Conference on Data Engineering (ICDE), pages 1332-1335, 2013.

[44]. S. Chiu and M. Shan. *Computer music composition based on discovered music patterns*. In 2006 IEEE International Conference on Systems, Man and Cybernetics, volume 5, pages 4401-4406, Oct 2006.

[45]. E. Miranda and J. Biles, editors. *Evolutionary Computer Music*. Springer, Berlin, 2007.

[46]. Josep Lluís Arcos, Xavier Serra, and Ramon Lopez de Mantaras. *Saxex : a case-based reasoning system for generating expressive musical performances*, Journal of New Music Research, 1997.

[47]. Josep Lluís Arcos and Ramon López De Mántaras. *An interactive case-based reasoning approach for generating expressive music*. Applied Intelligence, 14(1):115-129, Jan 2001.

[48]. Ramon Lopez de Mantaras and Josep Lluís Arcos. *AI and music from composition to expressive performance*.AI Mag., 23(3):43-57, September 2002.

[49]. Maarten Grachten, Josep Lluís Arcos, and Ramon López de Mántaras. *Tempoexpress, a CBR approach to musical tempo transformations*. In Peter Funk and Pedro A. González Calero, editors, Advances in Case-Based Reasoning, pages 601-615, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[50]. Taizan Suzuki, Takenobu Tokunaga, and Hozumi Tanaka. *A case-based approach to the generation of musical expression*. In Proceedings of the 16th International Joint

Conference on Artifical Intelligence - Volume 1,IJCAI'99, pages 642-648, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[51]. Carlos Eduardo and Cancino Chac'on. *A Computational Model of Immanent Accent Salience in Tonal Music*. PhD thesis, Doctoral Thesis, Johannes Kepler University, Linz, 2018.

[52]. Michele Della Ventura. *Computer system for designing musical expressiveness in an automatic music composition process*. In Jianyong Qiao, Xinchao Zhao, Linqiang Pan, Xingquan Zuo, Xingyi Zhang, Qingfu Zhang, and Shanguo Huang, editors, Bio-inspired Computing: Theories and Applications, pages 434-443, Singapore, 2018.Springer Singapore.

[53]. Sergio Giraldo and Rafael Ramírez. *A machine learning approach to ornamentation modeling and synthesis Injazz guitar*. Journal of Mathematics and Music, 10(2):107-126, 2016.

[54]. C. Liu and C. Ting. *Computational intelligence in music composition: A survey*. IEEE Transactions on Emerging Topics in Computational Intelligence, 1(1):2-15, Feb 2017.