Interactive Evolution of 8–bit melodies with Genetic Programming towards finding aesthetic measures for sound

Maximos A. Kaliakatsos–Papakostas[a], Michael G. Epitropakis[a],
Andreas Floros[b], and Michael N. Vrahatis[a]

[a]Computational Intelligence Laboratory, (CILab), Department of Mathematics,
University of Patras, GR-26110 Patras, Greece
{maxk,mikeagn,vrahatis}@math.upatras.gr
[b]Department of Audio and Visual Arts, Ionian University, GR-49100 Corfu, Greece
floros@ionio.gr

Abstract. The efficient specification of aesthetic measures for music as a part of modelling human conception of sound is a challenging task and has motivated several research works. It is not only targeted to the creation of automatic music composers and raters, but also reinforces the research for a deeper understanding of human noesis. The aim of this work is twofold: first, it proposes an Interactive Evolution system that uses Genetic Programming to evolve simple 8–bit melodies. The results obtained by subjective tests indicate that evolution is driven towards more user–preferable sounds. In turn, by monitoring features of the melodies in different evolution stages, indications are provided that some sound features may subsume information about aesthetic criteria. The results are promising and signify that further study of aesthetic preference through Interactive Evolution may accelerate the progress towards defining aesthetic measures for sound and music.

1 Introduction

Various systems have utilized *Interactive Evolution* (IE) through *Genetic Programming* (GP) [10] as a means for automatic music composition and sound synthesis. In parallel, an intense research effort [11, 12] is taking place for specifying a set of objective aesthetic criteria, if possible. The specification of aesthetic criteria for music and sound is important not only for the automatic composition of musical pieces, but also for the creation of efficient automatic music raters, i.e. systems that produce human–like evaluations of musical pieces automatically. Several works have provided interesting results towards this direction by examining symbolic features of musical compositions [13]. In parallel, evolutionary techniques have been used for sound synthesis but they have mainly focused on creating synthesized sounds that assimilate certain target sounds [4].

In [22, 23] two systems are presented where Genetic Algorithms (GA) and GP are combined to modify existing symbolic music compositions and create

novel ones. The potential of inducing objective aesthetic measures for symbolic compositions through subjective IE systems with the utilization of Artificial Neural Networks (ANNs) has been discussed in [9]. Some works have attempted to drive evolution with ANNs [13, 21] or Self Organizing Maps [14] trained on specified symbolic music features, as automatic fitness raters. The interested reader is referred to [2] for a review of symbolic music composition with genetic techniques.

The work at hand studies the field lying in the junction of IE and aesthetic measures for sound. Specifically, an IE system based on GP is presented which evolves functions belonging to a certain class and produce interesting sound output with structural coherence in multiple musical time scales. The evolution of these functions through an interactive scheme is indicated to be a promising methodology for exploring the sound characteristics that distinguish preferable melodies.

The motivation of the work at hand along with a brief description of these functions is presented in Section 2. Section 3 presents the IE system, its interface and refers to the features that we monitored as carriers of aesthetic information. In Section 4 the results obtained through subjective are presented, which show that the system leads the evolution towards more pleasing melodies and provide strong indications that some features may incorporate aesthetic meaning. The paper ends with conclusions and pointers to future work in Section 5.

2   Motivation and Background Material

For sound synthesis, GP has been used to evolve sinusoidal oscillators and filters to simulate a target sound [5]. An IE for sound synthesis has been utilized for the project described in [16]. This approach uses functions to directly shape waveforms, which are evolved according to fitness values provided by users. The functions used to directly sculpture waveforms *"produced little more than irritating noise and evolved (if at all) very slowly"* [16]. On the other hand, a class of functions that create waveforms with pleasant and interesting sonic output has recently emerged and is rapidly gaining attention among the retrograde 8–bit music movement [6, 7]. These functions have been mostly used to create 8–bit music, similar to the music created with Pulse Code Modulation (PCM) sound systems of early Personal Computer (PC) systems.

The sonic output of the aforementioned functions presents structural organization from the musical time scale of *micro* to *meso*, even sometimes intruding the borders of *macro*, as described in [17][1]. This organization allows the creation

---

[1] Time scales of music from *micro* to *macro* as defined in [17]: *Micro*: Sound particles on a time scale that extends down to the threshold of auditory perception (measured in milliseconds). *Sound object*: A basic unit of musical structure, generalizing the traditional concept of note to include complex and mutating sound events on a time scale ranging from a fraction of a second to several seconds. *Meso*: Division of form. Grouping of sound objects into hierarchies of phrase structures of various

---

Algorithm 1 Construction of an 8–bit waveform of 8000Hz sample rate through a function $f(t)$

Input: i) A functional expression $f(t)$ and **ii)** time duration in seconds (d)
Output: The waveform of an audio signal with d seconds duration
 1: **for** $t = 1$ **to** $d\,8000$ **do**
 2:    **if** $f(t) ==$NaN **then**
 3:        $q(t) \leftarrow 0$
 4:    **else**
 5:        $q(t) \leftarrow \mathrm{mod}(f(t), 256)$
 6:    **end if**
 7:    $s(t) \leftarrow 2\frac{q(t)}{255} - 1$
 8: **end for**

---

of an IE system that attracts the attention of the human rater (user). Consequently, subjective evaluations can be provided regarding multiple musical time scales, thus enabling us to measure aesthetic aspects of sound that were previously undetected. Furthermore, a deeper exploration of the "compositional" potential of these functions can be realized through GP.

The aim of this work is twofold. The first aim is to create an IE system that evolves functions which create sounds organized in melodic parts through GP. Consequently, this aim may lead to a report on results about the challenges faced towards this direction. Secondly, we aim at providing a first analysis on the aesthetic potential that some sound features may incorporate. These features move beyond *sound object* and below *meso* time scales, thus combining the analysis performed so far on two different fields: synthesized notes and automatic algorithmic compositions of symbolic music respectively.

The class of functions which create waveforms with structural coherence in many levels, from sound texture to musical composition, has recently emerged and rapidly gained the attention of many programmers–composers [6]. These functions have mainly been used to create 8–bit compositions, producing music content equivalent to the early PCM digital coding in terms of sampling frequency and quantization resolution. Also, these functions may have arbitrarily many variables and besides the standard arithmetic operators ("+", "−", "∗", "/") they also use operators defined within the C language syntax. In this work however, we use a single variable that is the most usual case in experimental applications of these functions so far. Furthermore, we have experimented with the subset of the available C operators which are logical bitwise AND (&), OR (|), XOR (^), bitwise left shift (<<) and bitwise right shift (>>). For a thorough analysis on the sound properties of these functions, the interested reader is referred to [7].

The construction of the waveforms through the examined functions is described in Algorithm 1 while a graphical example is given in Figure 1. As men-

---

sizes, measured in minutes or seconds. *Macro*: The time scale of overall musical architecture of form, measured in minutes or hours, or in extreme cases, days.

tioned previously, these waveforms have an 8–bit resolution and a sample rate of 8000Hz. Outlining the way that the waveforms are created, we set an integer counter, t, that takes values between 1 and $d\,8000$ and represents the generated music sample indices. d is the desired duration of the sound output in seconds. Then we evaluate the functional expression, $f(t) \in \mathbb{Z}$, for every $t \in \{1, 2, \ldots, d\,8000\}$. During the computation of $f(t)$, division by zero is assumed to provide a value of 0. Since we consider 8–bit audio depth the available digital audio sample levels are $2^8 = 256$, thus we simulate the wrapping overflowing behavior of 8–bit computer systems by forming the "*quantized*" sequence $q(t) = \mod(f(t), 256)$ for $t \in \{1, 2, \ldots, d\,8000\}$. Finally, the waveform $s(t)$ is calculated by normalizing $q(t)$ in the range $[-1, 1]$ by $s(t) = 2\,(q(t)/255) - 1$.
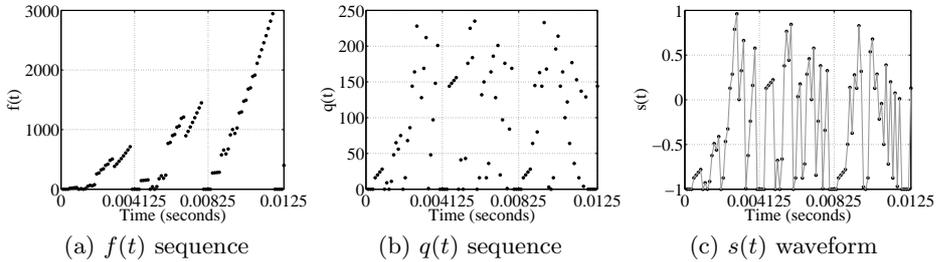

| (a) $f(t)$ sequence | (b) $q(t)$ sequence | (c) $s(t)$ waveform |

Fig. 1. Example of the transformation of the sequence `f(t) = t*(t>>8*(t>>15|t>>8)&(20|(t>>19)*5>>t|t>>3))`, `t∈` $\{1, 2, \ldots, 100\}$ to $q(t)$ and finally to the waveform $s(t)$.

3   The proposed approach

In this section we explore the potential of extracting possible features towards finding aesthetic measures through an interactive evolution system. Three types of features were examined as carriers of information that would expose aesthetic meaning. These three categories are *Waveform Information*, *Spectral and Cepstral Features* and *Tree–based Features*. With waveform information we measure the information capacity of the quantized sequences that form the final waveform. With spectral and cepstral features we monitor some frequency domain aspects which may incorporate aesthetic meaning. For additional information about these features the interested reader is referred to [24]. Finally, with tree–based features we intend to capture structural characteristics of the tree representation of functions, regarding size and operators. Especially with *operator entropy* we try to capture diversity and homogeneity of the operator distribution. An analysis on whether or not some operations produce more "pleasing" art has previously been presented in [3], with an analysis on automatically created images. A short description of the aforementioned features along with their acronyms is provided in Table 1.

Table 1. The features examined as carriers of aesthetic information

Waveform information	
Fractal Dimension (**FD**)	Fractal dimension of the quantized sequence with the Higuchi [8] algorithm
Shannon Information Etropy (**SIE**)	Shannon Information Entropy [19] of the normalized (to unit sum) histogram of the quantizes sequence
Compressibility through compression rate (**CR**)	Ratio of the size of the compressed quantized sequence with the Lempel–Ziv algorithm [26] over the size of the uncompressed sequence
Spectral and Cepstral features [24]	
Spectral Centroid (**SC**)	The "*center of weight*" of the spectrogram
Spectral Centroid Standard Deviation (**SCstd**)	Standard deviation of the the spectral centroids within short time segments (of 0.1299 seconds)
Mean Spectral Flux (**SFm**)	The mean value of spectral fluxes (Euclidean distances of the spectrogram of short consecutive segments) of segments of 0.1299 seconds
Spectral Flux Standard Deviation (**SFstd**)	The standard deviation of the aforementioned spectral fluxes
Spectral Roll–off (**SR**)	The frequency below which the 85% of the total energy of the spectrogram is concentrated
Spectral Roll–off Standard Deviation (**SRstd**)	Standard Deviation of the Spectral Roll–offs of short consecutive time segments (of 0.1299 seconds).
Mel–Frequency Cepstral Coefficients Entropy (**MFCCe**)	the Shannon Information Entropy of the normalized (to unit sum) histogram of the Mel–Frequency Cepstral Coefficients
Tree–based features	
Maximum Tree Depth (**MTD**)	The maximum depth of the tree representation of an individual
Number of Tree Nodes (**NTN**)	The number of nodes in the tree representation of an individual
Operator Probability Density Function (PDF) (**OpPDF**)	The PDF of the operators in an individual
Operator PDF Entropy (**OpE**)	The Shannon Information Entropy of the OpPDF of each individual.

The IE system uses GP to evolve the functional expressions discussed in Section 2. The fitness value of each individual is provided by the user, after hearing the sound output (phenotype) it produces for as long as she/he wishes. Some sound outputs have interesting melodic content, with several melodic and rhyth-

mic variations, while others produce rather uninteresting music forms, with dull repetitions. Since the user may not be sure about the variation potential of each melody, she/he should spend a considerable amount of time hearing repeating motifs, a fact that increases fatigue. For this reason, several visualizations are provided in parallel with the sound playback so that the user may anticipate the expected variation potential of the individual she/he hears. These visualizations include the spectrogram, the Mel–Frequency Cepstral Coefficients (MFCCs) and the plot of the quantized sequence ($q(t)$) among others. Figure 2 illustrates a screen shot of the visualizations that are produced during the playback of an individual.
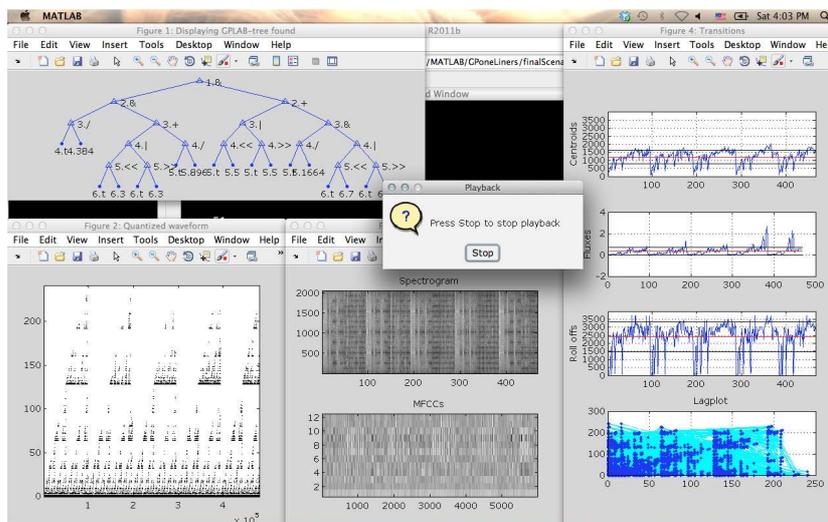

Fig. 2. Screen shot of the visualizations during the playback process.

The evolutionary process follows the standard GP methodology. The individuals of the current population are going through a selection stage, where the parents of the next generation are specified. Three versions were created with different selection schemes, two of which are the standard *roulette* and *tournament* methods. For a thorough description of the genetic operators and the selection procedures mentioned so far, the reader is referred to [15]. The third selection scheme was an extreme methodology that we call "*elitist*", where only the fittest individuals among the current and former generation in every step were selected as parents. The motivation behind this approach was towards reducing the user fatigue caused by emerging offspring that produced rather noisy or uninteresting phenotype. The tradeoff was the reduced potential of extremely novel audio content. User fatigue in IE systems, and especially in systems that produce art, is an important factor since it does not only affects the user's engagement to the rating task, but may consequently mislead the evolutionary process [25].

The selected individuals provide offspring for the next generation through the *subtree crossover* and the *subtree mutation* operators, also known as *headless chicken crossover* [1] operator. The genetic operator was selected randomly with crossover having 0.9 and mutation 0.1 probability being chosen. After experimenting with the genetic operators, we observed that individuals of extremely small and large depth tend to produce uninteresting and noisy sound respectively. For this reason we employ depth constraints to the offspring by re–performing the selected operation with the selected individuals until the depth of their offspring was between 3 and 10.

Population initialization was firstly designed to be random, but the produced individual phenotypes were most commonly uninteresting or noisy. Such an initialization would hardly produce interesting findings, since the rating procedure would just discard noisy individuals. The initial population was thus chosen to be constituting of individuals that are randomly chosen among a set of predefined individuals with certified melodic attributes. In this way, the user is able to drive the evolutionary process towards her/his subjectively chosen direction. To the presented IE system, each user is able to select the number of individuals in each generation. For the presented results however, the users were advised to choose 4 individuals, which seems to be a good compromise between population diversity and evolution potential of the proposed IE system. Finally, the implementation of the IE system was developed in MATLAB using a modified version of "*GPLAB*" [20]. In order to make the system as easy–to–use as possible, the communication between the user and the system was performed through a Graphical User Interface (GUI).

4 Results

The results presented in this work comprise of statistics gathered among 10 participants–users on their first trial of the IE system. No record with personal data about the users was kept, but we can mention that most of them (7 out of 10) were playing a musical instrument, from whom the 4 had at least 5 years of music education. The participants were not aware of the purposes of the research, and were not informed about the way that system functioned before they started their trial. The only information they were provided had to do with their interaction with the system through the GUI. Before the beginning of each trial, each user heard three sample melodies to get accommodated with the music style and sound textures of the 8–bit melodies. They were advised to rate each melody according to their taste and to feel free to quit the program any time they liked. The hearing process was controlled by the user, who was prompted to stop the melody any time she/he wished. Before the beginning of each trial, the users were advised to consult the visualizations for determining the alteration potential of the melody they heard. After hearing each melody, a rating dialog was appearing prompting the user to rate the melody just heard. The rating scale was the integer values between 0 and 10, with 0 being the worse,

and the participants were advised to freely rate the individuals according to their personal taste.

4.1   Convergence to subjective optima

We have divided the participants in three groups so that all three selection versions of the system were tested. Thus, 3 participants used the *"Elitist"* version, 4 participants used the *Roulette* and 3 the *Tournament* version. Table 2 demonstrates the overall improvement of the user ratings from the initial to the last population each user rated. The FI and FL indices are used when we refer to fitness ratings in the initial and the last population respectively, while the GN indicates the number of generations. The mean value of the aforementioned quantities is denoted μ, their standard deviation by σ, while max and min denote their maximum and minimum values respectively. The relative fitness change of the means from the initial to the last population is denoted with r_f, hence $r_f = (\mu_{FL} - \mu_{FI})/\mu_{FI}$.

The positive value of r_f for every version shows that the mean fitness value increased from the initial to the final population, which reveals that the proposed system captures, at some extent, the subjective aesthetics of the user. The best relative improvement is recorded for the *Tournament* version with 0.543 ratio. It has to be noted though that one of the three users of the *"Elitist"* version evolved one generation, which he rated worst than the initial population and then quit the trial. The maximum mean number of generations was also achieved for the *Tournament* version. The standard deviation of the ratings in the final population is decreased in relation to the initial population, which means that the fitness of all individuals in the final population are close.

Table 2. Results for the improvement of ratings between the initial and final populations for all three versions of the IE system.

	μ_{GN}	min$_{FI}$	μ_{FI}	σ_{FI}	max$_{FI}$	min$_{FL}$	μ_{FL}	σ_{FL}	max$_{FL}$	r_f
"Elitist"	4.333	0	**4.417**	2.151	7	2	**5.167**	2.125	9	**0.170**
Roulette	6	0	**3.438**	2.190	7	1	**4.250**	1.880	7	**0.236**
Tournament	9	0	**2.917**	2.811	7	1	**4.500**	1.679	7	**0.543**

4.2   Towards aesthetic measures for sound

The melodies of the individuals that were candidates as members of the initial population were collected from the internet [6]. These melodies have been constructed by programmers–electronic music composers with experimentation on possible combinations of operators and constants. In the previous paragraphs we observed that the evolution of these melodies with GP produced new ones which were more preferable by the users. We try to capture the impact of this

shift towards more preferable sounds by monitoring the change in the features referred in Table 1. Table 3 demonstrates the mean value of the waveform, spectral and cepstral features, as denoted in Table 1, of the individuals in the initial and the last population for the trials of all users. Furthermore, it exhibits their relative change, which is also illustrated in Figure 3 (a). The relative changes of the tree–based features are depicted in Figure 3 (b) and (c). The relative change of a value from step i, v_i, to step j, v_j, is defined as $r = (v_j - v_i)/v_i$.

Table 3. The mean values of the waveform, spectral and cepstral features for the melodies in the initial and the last population of the trial of all user and their relative changes. The features that have a relative change above 0.2 in magnitude are marked with boldface numbers.

	Waveform info			Spectral and Cepstral features						
	FD	SIE	CR	SC	SCstd	SFm	SFstd	SR	SRstd	MFCCe
Init. pop.	1.345	3.628	0.030	1083	261.3	1.85	1.105	2004	476.4	3.012
Last pop.	1.635	2.581	0.020	998.2	352.4	0.51	0.427	1894	659.2	3.036
Rel. change	**0.216**	**-0.289**	**-0.333**	-0.078	**0.349**	**-0.724**	**-0.614**	-0.055	**0.384**	0.008

The small number of participants does not allow for safe conclusions to be drawn. However, the presented results are evidential about the potential impact of some features on the aesthetic properties of sound, since their relative change is considerable. For example, the spectral flux mean and standard deviation (SFm and SFstd) decreased impressively, with a parallel considerable increase in the standard deviation of spectral centroids (SCstd). This fact reveals that the intense spectral modulations within 0.1299 seconds, which are captured by spectral flux, are not pleasant. The increase in SCstd on the other hand, shows that more intense spectral modulations are preferred, but in a larger time scale. The spectral roll-off features (SR and SRstd) could probably be discarded, since they follow the behavior of the spectral centroid features.

Interesting results were also provided by the information entropy and complexity measures. The compression rate (CR) decreased, indicating that waveforms which exhibit less repeating patterns are preferred. On the other hand, the Shannon Information Entropy (SIE) of the histogram of the quantized sequence ($q(t)$) that formed the waveform was reduced, outlining "sharper" histograms, with less amplitude bins being used. The fractal dimension (FD) was also increased to 1.635, which is a value approximating *golden ratio* (1.618). The relation of symbolic music characteristics and the golden ratio has also been noticed in [12]. By combining these results, one could claim that there is a structural organization of music in multiple levels which may be expressed by the golden ratio.

Figure 3 (c) illustrates that tree characteristics like the mean tree depth (MTD), the mean number of tree nodes (NTN) and the operator entropy (OpE) did not present important change. Figure 3 (b) demonstrates that the bitwise "right shift" (>>) and the bitwise "and" (&) operators were more preferable, while

the utilization of arithmetic operators was reduced. The -1 value in the relative change of the bitwise "left shift" (<<) operator occurred because there was only one such operator in the initial population among all trials, which diminished. Finally, Figure 3 (d) depicts an example of a user rating behavior, i.e. the circles in the graph correspond to the mean rating value at each generation, while the error bars around the mean depict the standard deviation. This behavior is similar among most of the users. Specifically, at the first stages of evolution the ratings decreased, before increasing again to reach high rating values at the last stages. Some example melodies together with information about the generation they appeared and the fitness value they were provided by a participant during a trial can be found at [18].
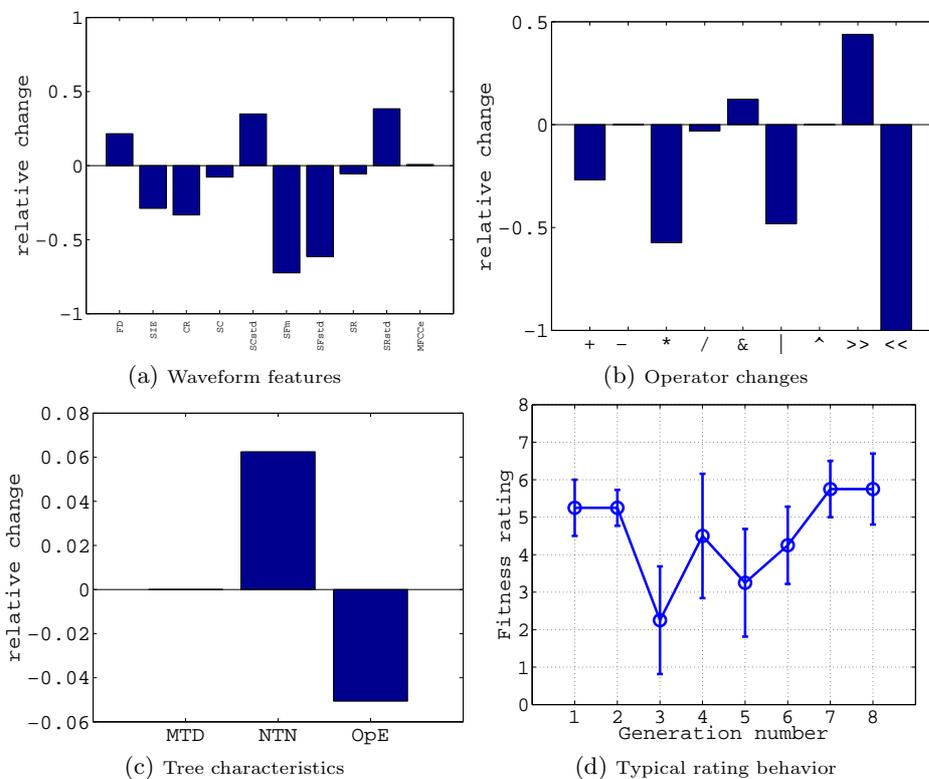

(a) Waveform features

(b) Operator changes

(c) Tree characteristics

(d) Typical rating behavior

Fig. 3. (a) Relative change of waveform, spectral and cepstral features, (b) relative operator changes, (c) relative change of tree characteristics and (d) a typical rating example of a user for each generation, with circles depicting the mean rating and error bars their standard deviation.

5   Conclusions

This work presents an Interactive Evolution (IE) system that uses Genetic Programming (GP) to evolve 8–bit melodies produced by a certain class of functions. Moreover, a discussion about the potential use of this system for capturing aesthetic criteria for sound was carried out. Three versions were tested that included different selection schemes. Results are reported on trials performed by 10 participants on all three versions. The participants, being not aware of the aims of this research, rated the individuals–melodies in all generations subjectively. The results show that all three versions evolved the initial melodies to new ones that were more pleasing for the majority of the participants. Towards establishing aesthetic measures for sound, we examined features regarding the waveform, spectral, cepstral and tree–based characteristics of the individuals in each generation. The results indicate that this direction could produce fruitful findings about musical structure and sound in many levels.

As a future work, we initially intent to create a web platform for this IE system in order to make it accessible to many participants. Additional information about each participant will also be requested, including musical experience, age and education among others. Such an analysis would also provide insights about the connection of aesthetic criteria to the experiences of a person. Furthermore, future work should also include experimentation on making the evolutionary process more effective and interesting for the user. For example, we could employ a similarity testing procedure between the phenotypes of offspring and the population in the gene pool to prevent the appearance of extremely similar individuals from generation to generation. Finally, we could amplify the interactive part of the evolutionary process by letting the user decide about certain aspects of the evolution, from the choice of the genetic operators to be applied to the selection of individual(s) to be genetically modified.

References

1. Angeline, P.J.: Subtree crossover: Building block engine or macromutation? In: Genetic Programming 1997 Proceedings of the Second Annual Conference. pp. 9–17. Morgan Kaufmann (1997)
2. Burton, A.R., Vladimirova, T.: Generation of musical sequences with genetic techniques. Computer Music Journal 23(4), 59–73 (1999)
3. Ekárt, A., Sharma, D., Chalakov, S.: Modelling human preference in evolutionary art. In: Proceedings of the 2011 international conference on Applications of evolutionary computation - Volume Part II. pp. 303–312. EvoApplications'11, Springer-Verlag, Berlin, Heidelberg (2011)
4. Garcia, R.A.: Towards the automatic generation of sound synthesis techniques: Preparatory steps. In: Audio Engineering Society Convention 109 (Sept 2000)
5. Garcia, R.A.: Growing sound synthesizers using evolutionary methods. Synthesis M(Garcia 2000), 99–107 (2001)
6. Heikkilä, V.: countercomplex–bitwise creations in a pre-apocalyptic world. http://countercomplex.blogspot.com/ (Dec 2011)

7. Heikkilä, V.: Discovering novel computer music techniques by exploring the space of short computer programs. arXiv:1112.1368 (Dec 2011)
8. Higuchi, T.: Approach to an irregular time series on the basis of the fractal theory. Phys. D 31, 277–283 (June 1988)
9. Johanson, B.E., Poli, R.: GP-Music: an interactive genetic programming system for music generation with automated fitness raters. Tech. Rep. CSRP-98-13, University of Birmingham, School of Computer Science (May 1998)
10. Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. The MIT Press, 1 edn. (Dec 1992)
11. Manaris, B., Purewal, T., McCormick, C.: Progress towards recognizing and classifying beautiful music with computers - MIDI-encoded music and the Zipf-Mandelbrot law. In: Proceedings IEEE SoutheastCon, 2002. pp. 52–57. IEEE (2002)
12. Manaris, B., Romero, J., Machado, P., Krehbiel, D., Hirzel, T., Pharr, W., Davis, R.B.: Zipf's law, music classification, and aesthetics. Computer Music Journal 29(1), 55–69 (Apr 2005), ArticleType: research-article / Full publication date: Spring, 2005 / Copyright 2005 The MIT Press
13. Manaris, B., Roos, P., Machado, P., Krehbiel, D., Pellicoro, L., Romero, J.: A corpus-based hybrid approach to music analysis and composition. In: Proceedings of the 22nd national conference on Artificial intelligence - Volume 1. pp. 839–845. AAAI Press (2007)
14. Phon-Amnuaisuk, S., Law, E., Kuan, H.: Evolving music generation with som-fitness genetic programming. In: Giacobini, M. (ed.) Applications of Evolutionary Computing, Lecture Notes in Computer Science, vol. 4448, pp. 557–566. Springer Berlin / Heidelberg (2007)
15. Poli, R., Langdon, W.B., McPhee, N.F.: A Field Guide to Genetic Programming. Lulu Enterprises, UK Ltd (Mar 2008)
16. Putnam, J.B.: Genetic programming of music (30 August 1994)
17. Roads, C.: Microsound. MIT Press (2004)
18. Sample of melodies–individuals created during a trial by a participant: https://sites.google.com/site/maximoskp/SampleMelodies.zip (2012)
19. Shannon, C.E.: A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review 5, 3–55 (January 2001)
20. Silva, S., Almeida, J.: Gplab-a genetic programming toolbox for matlab. Proceedings of the Nordic MATLAB conference pp. 273–278 (2003)
21. Spector, L., Alpern, A.: Induction and recapitulation of deep musical structure. In: Proceedings of the IFCAI-95 Workshop on Artificial Intelligence and Music. pp. 41–48 (1995)
22. Spector, L., Alpern, A.: Criticism, culture, and the automatic generation of artworks. In: Proceedings of the twelfth national conference on Artificial intelligence (vol. 1). pp. 3–8. AAAI '94, American Association for Artificial Intelligence, Menlo Park, CA, USA (1994)
23. Tokui, N.: Music composition with interactive evolutionary computation. Communication 17(2), 215226 (2000)
24. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Transactions on Speech and Audio Processing 10(5), 293–302 (Jul 2002)
25. Yan, J.R., Min, Y.: User fatigue in interactive evolutionary computation. Applied Mechanics and Materials 48-49, 1333–1336 (Feb 2011)
26. Ziv, J., Lempel, A.: A universal algorithm for sequential data compression. IEEE Transactions on Information Theory 23(3), 337– 343 (May 1977)