INSIGHTS IN HABITS AND ATTITUDES REGARD-ING PROGRAMMING SOUND SYNTHESIZERS: A QUANTITATIVE STUDY

Gordan Kreković
Visage Technologies
gordan.krekovic@visagetechnologies.com

ABSTRACT

Sound synthesis represents an indispensable tool for modern composers and performers, but achieving desired sonic results often requires a tedious manipulation of various numeric parameters. In order to facilitate this process, a number of possible approaches have been proposed, but without a systematic user research that could help researchers to articulate the problem and to make informed design decisions. The purpose of this study is to fill that gap and to investigate attitudes and habits of sound synthesizer users. The research was based on a questionnaire answered by 122 participants, which, beside the main questions about habits and attitudes, covered questions about their demographics, profession, educational background and experience in using sound synthesizers. The results were quantitatively analyzed in order to explore relations between all those dimensions. The main results suggest that the participants more often modify or create programs than they use existing presets or programs and that such habits do not depend on the participants' education, profession, or experience.

1. INTRODUCTION

During the last five decades, sound synthesis strongly contributed in shaping the path of music evolution. The technology that allowed creating an endless variety of novel sounds brought greater freedom in expressing musical ideas and encouraged musicians to be more innovative and ambitious in their artistic intentions. In order to increase flexibility of sound creation, synthesizers typically provide musicians with a large number of controllable parameters. However, since synthesis parameters do not necessarily bear acoustical meaning and they can depend on each other, managing numerical parameters is a difficult and time-consuming activity which can negatively affect inspiration and productivity [1].

In order to mitigate this problem, researchers have proposed solutions based on automatic selection of synthesis parameters which allow musicians to create desired sounds more intuitively. Instead of controlling numerical parameters manually, musicians can define their requirements in several other ways: (1) by providing a sound sample perceptually similar to the target sound, (2) by describing the target sound by using attributes (such as bright and harsh), and (3) by using more intuitive interfaces (such as visualizations of timbre spaces and scoring of automatically generated sounds). Mapping those inputs or actions into synthesis parameters is a non-trivial problem that is usually approached using various computer science techniques.

Automatic selection of sound synthesis parameters is a relevant research challenge, especially nowadays when artificial intelligence is starting to emerge as a mean of advanced automation. Besides being academically interesting, automatic parameter selection has the potential to change the way how musicians use sound synthesizers. However, although this practical research topic is primarily motivated by possible pragmatic improvements, it has not been informed or guided by user experience (UX) studies. While the research has been ongoing for almost three decades, existing solutions are still not widely accepted in practical use and they are scattered across a variety of approaches and problem definitions.

A comprehensive study on attitudes and habits of musicians who use sound synthesizer might help in articulating the research question in terms of defining which specific problems automatic parameter selection should address. It may also help in explaining and assessing the relevance of different problem definitions and possible technical approaches. Thorough understanding of users' needs, habits, and attitudes will help researchers opt for design decisions which maximize usability and usefulness of their solutions. Insights about the practical context are also relevant for theoretical studies, which do not aim for applicability, but for demonstrating novel ideas and concepts, because the practical context provides the realistic expectations and allows the explicit ratio between theoretical knowledge and applicability. Finally, a study on users` habits and attitudes can also inform the process of designing new interfaces for sound synthesizers or at least serve as a starting point for further user research. Although a few UX studies related to sound synthesizers exist [2, 3] they are not fully aimed at informing the research on automatic parameters selection.

In light of that, the purpose of this paper is to investigate attitudes and habits of musicians who use software or hardware sound synthesizers. The approach is explorative and it also takes into account users' demographic data, profession, educational background and experience in using sound synthesizers, relating those dimensions to their habits and attitudes. To reach this objective, we conducted a questionnaire that included 122 participants and quantitatively analyzed results.

The paper starts with a brief overview of certain previous studies on automatic selection of synthesis parameters, which form an appropriate context of this research. The rest of the paper explains methodology, results, discussion, and conclusions of the quantitative research on programming sound synthesizers.

2. RELATED WORK

A considerable number of studies related to automatic selection of synthesis parameters emerged in the interdisciplinary field of computer music technology during the last three decades. Those studies explored three aforementioned ways of defining a desired sound: (1) by providing a perceptually similar sound, (2) by describing it using attributes, and (3) by interacting with a more intuitive user interface.

Matching a provided perceptually similar sound can be observed as an optimization problem in the space of synthesis parameters with the aim to minimize the perceptual difference between the provided sound and the synthesized result. Evolutionary algorithms are an appropriate approach to solving such kind of optimization problems, so they were the primary choice of many researchers, including the pioneers Andrew Horner and his colleagues. After reporting successful results with an FM synthesizer [4], they conducted similar studies for other sound synthesis techniques using the fitness function based on the same similarity measure – the absolute difference of two discrete Fourier transforms [5-7]. Some researchers noticed shortcomings of the selected similarity measure and proposed their solutions taking into account psychoacoustic phenomena [8-10]. Target matching using evolutionary algorithms with automatic calculation of the fitness function have been extensively studied and applied to various synthesis techniques including additive synthesis [11], subtractive synthesis [12], noise shaping [13], granular synthesis [14], plucked string synthesis [8], dynamic stochastic synthesis [15], and even synthesizers with multiple synthesis engines [16]. In addition to similarity measures calculated from the signal, several authors explored interactive evolutionary algorithms and proposed solutions that rely on the fitness values provided by the user [17-19].

Besides evolutionary algorithms, some studies explored and applied other computer science techniques such as fuzzy logic [20] and deep neural networks [21].

Marginally related to the problem of target matching are feature synthesizers capable of producing sounds from a given set of audio features that are either extracted from a target sound or provided by the user [22-26].

In contrast to automatic target matching, only several authors focused on controlling sound synthesizers using timbral attributes. Miranda presented a system based on decisions trees used to induce relations between quasi-timbral attributes and synthesis parameters [1], while Gounaropoulos and Johnson employed a neural network to learn relations between adjectives and audio features of a sound characterized by those adjectives [27]. Another approach is decomposing the inherently complex problem into two simpler steps: the first one is mapping timbral attributes into audio features using an expert system based on fuzzy logic, while the second step is a pseudo-heuristic search for appropriate synthesis parameters to match the target audio features [28].

Some other notable solutions include: a knowledge-based system for controlling FM synthesizers [29], a system for sound synthesis and transformation based on adjectives, SeaWave [30], keyword analysis and clustering [31], an expert system for mapping adjectives directly to sound synthesis parameters [32], and an interactive evolutionary algorithm extended with adjective control [33].

In most of the aforementioned literature, the focus was on technical solutions without detailed explanations of the problem from the users` point of view. Moreover, it is rarely clear how the proposed solutions are intended to be used – as a tool that supports users when creating new sounds, as a tool that completely offloads users from that type of work, as a tool for inspiring musicians, or something else. For that reason, the aim of this paper is to strengthen the user dimension and draw more attention to user experience.

3. METHODOLOGY

Quantitative results presented in this paper are obtained by analyzing responses to a questionnaire about habits and attitudes of sound synthesizer users when creating, modifying, and using sounds. The questionnaire, which mostly consisted of questions with predefined ordinal and categorical answers, was divided into two sections.

The first section included the questions about participants' demographics (gender and age), primary field of work or education (since some synthesizer users might not be professional musicians or music students), level of music education, and experience with sound synthesizers (i.e. duration of use).

The second section dealt with participants' habits and attitudes regarding using, modifying, and creating programs in sound synthesizers. The questions from the second section were about (1) their tendencies to use predefined or existing programs, modifying existing programs, and creating new programs from scratch, (2) actions the participants are likely to take when the desired sound is not predefined, (3) impediments of creating and modifying programs manually, (4) features of sound synthesizers that can help them most in creating and modifying programs, and (5) potential helpfulness of hypothetical functions for automatic or semiautomatic selection of synthesis parameters. Most of the questions consisted of a common part (e.g. "How often do you take the following actions when using sound synthesizers?") and a specific statement (e.g. "Using predefined programs from the synthesizer without modifications", "Creating your own programs from scratch") treated as an ordinal question with a 5-point Likert scale (e.g. "never", "rarely", "sometimes", "very often", "always"). Besides the questions with predefined ordinal or categorical answers, there was also an optional question about the challenges that users face when creating or modifying programs. The whole questionnaire is available here:
https://goo.gl/forms/3Tc7XBolkLjzr19k2

When the questionnaire was ready and validated through test runs, it was disseminated using Facebook groups, Internet forums, and direct contacts. One Facebook group was more oriented towards researchers, while all other groups and forums were general user groups not focused on any particular music creation tools, brand, or product of music industry.

The quantitative analysis of the collected responds included calculating statistics for each question, as well as analyzing pairs of answers. Appropriate statistical tests were selected based on answer types: Wilcoxon rank-sum test for pairs of categorical and ordinal answers, chi-squared hypothesis tests for pairs of categorical answers, and Spearman's rank correlation for correlations between ordinal answers.

Such quantitative research methods are typically used in the field of human-computer interaction and the same methodology was applied to some topics related to music [34-36].

4. RESULTS

4.1 About Participants

The demographic structure of the participants suggests a significant gender bias with 95.1% of male participants and only 2.5% of female participants. Since the responses were collected using a non-discriminatory, unbiased, and anonymous method, this statistic may indicate a gender bias in the field, similar to disproportions found in other technical domains [37]. The mean age of participants is 41.3, while the standard deviation is 12.2.

As their primary field of professional work or formal education, most of the participants stated computers and technology (28.2%), music (28.2%), arts and communications (12.1%), and management, business and finance (9.7%). Regarding their educational background in music, one participant had no education in music, 28.2% are self-taught, 17.7% had some training or lessons, 11.3% have basic music education (elementary music school, preparatory school, etc.), 23.4% have an advanced formal education, while 17% completed a conservatory or academy (Bachelor or Masters of Music and higher) as shown in Figure 1. As expected, the Wilcoxon rank-sum test confirmed that participants who are professional musicians or music students have significantly higher levels of music education than the others (p<.01).

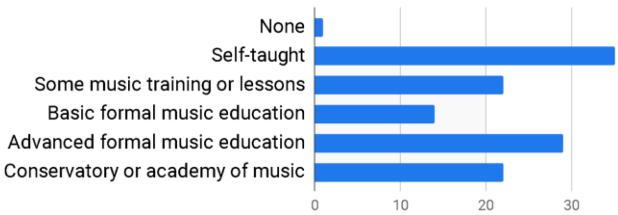

Figure 1. Participants' music education levels.

In general, the participants have a lot of experience with using sound synthesizers, since 71% of them have been using sound synthesizers for more than 10 years, 14.5% between three and ten years, 9.7% between one and three years, and 2.4% between three months and a year, while the remaining 2.4% are novice users with less than 3 months of experience. According to the Spearman's correlation coefficient, there is a very weak positive correlation between the participants' experience and the level of music education (r_s=.14, p<.01), and a moderate positive correlation between the experience and their age (r_s=.48, p<.01).

4.2 Usage Habits

In order to ascertain the usage habits, the participants were asked to state how often they take the following actions when using sound synthesizers: 1) using predefined programs (i.e. presets) without modifications (activity *A1*), 2) using existing programs created by others without modification (*A2*), 3) modifying predefined or existing programs (*A3*), and 4) creating programs from scratch (*A4*). For each activity, the possible answers were based on a Likert frequency scale. The distributions per action are shown in Figure 2. The results of the Wilcoxon rank-sum tests between all pairs of activities indicate that the participants more often modify existing programs (*A3*, Median=*Very often*) or create new programs from scratch (*A4*, Median=*Very often*) than they use presets (*A1*, Median=*Sometimes*) or existing programs without modification (*A2*, Median=*Sometimes*). All Wilcoxon rank-sum tests conducted between the pairs *A1-A3*, *A1-A4*, *A2-A3*, and *A2-A4* confirmed statistically significant differences between answers (p<.01 for all the aforementioned pairs), while no such differences were found between *A1-A2* and *A3-A4*. These observations have been made considering all the participants as one group, but the same results have been obtained on specific subgroups: the participants with less than 3 years of experience with sound synthesizers, the participants who are not professional musicians or music students, and even the participants without formal music education (i.e. those without any education, self-taught participants, and those who had some trainings or lessons).

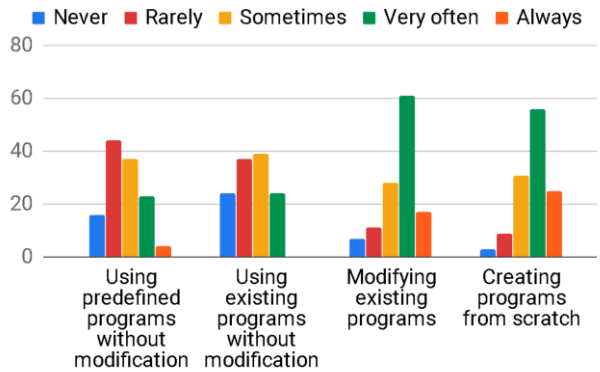

Figure 2. Frequencies of taking different actions when using sound synthesizers.

The indication that the aforementioned habits are neither related to the participants' experience nor their music education level has been confirmed by calculating Spearman's correlation coefficients between those dimensions. All the values of the obtained coefficients were between -0.1 and 0.1 with p>.5.

The next question in the survey also referred to the usage habits. The participants were asked how likely they would take the following actions if they needed a sound that was not included in the presets: 1) use another synthesizer that might have such a program (action *B1*), 2) search for an appropriate program for their synthesizer (e.g. online) (*B2*), 3) modify one of the presets (*B3*), and 4) create their own program from scratch (*B4*). The results suggest that the participants are in general least likely to search for an appropriate program (*B2*, Median=*Unlikely*), undecided when it comes to using another sound synthesizer (*B1*, Median=*Undecided*), and likely to modify one of the presets (*B3*, Median=*Likely*) or create new programs from scratch (*B4*, Median=*Likely*). The series of Wilcoxon rank-sum tests between the pairs of answers confirmed that there are statistically significant differences between *B1-B2*, *B1-B3*, *B1-B4*, *B2-B3*, and *B3-B4* (p<.01, for all the pairs), while there is no significant difference between *B3-B4*. These findings have been made by considering all participants, but the same results have been obtained for the subgroup of the participants without formal music education and those who are not professional musicians or music students.

The Spearman's correlation coefficients show that the likelihood of taking aforementioned actions is neither monotonically correlated with the participants' experience nor their music education level, because all of the coefficient values were between -0.1 and 0.1 with p>.5.

4.3 Impediments of Synthesizer Programming

Understanding impediments is as equally important as understanding usage habits. The participants were asked to express their level of agreement on a five-point Likert scale with the following statements about impediments of creating and modifying programs manually: 1) it can be time consuming (impediment *I1*), 2) it can distract them from focusing on music (*I2*), 3) it can be difficult and not intuitive to learn how to use a particular synthesizer (*I3*), and 4) it rarely leads to the desired results (*I4*). The participants in general agreed with the statements about the time consumption (*I1*, Median=*Agree*), distraction (*I2*, Median=*Agree*), and lack of intuitiveness (*I3*, Median=*Agree*), but disagreed with the last statement (*I4*, Median=*Disagree*). The same results have been obtained for all participants, but also for the specific subgroups: the participants with less than 3 years of experience, the participants who are not professional musicians or music students, and those without formal music education.

A weak negative monotonic correlation has been found between the statement *I4* and the participants' experience (Spearman's correlation: r_s=-0.30, p<.01) generally suggesting that the longer the participants use the synthesizers, the less they agree that manual programming rarely leads to the desired results. An even more interesting finding is a weak positive monotonic correlation between the education level and the statement *I3* (Spearman's correlation: r_s=0.25, p<.01). The higher education participants have, the more they agree that it can be difficult and not intuitive to use a particular sound synthesizer.

Regarding the relations between habits and impediments, the participants who use presets without modifying them more often (*A1*), rated all the statements about impediments with generally higher points. The Spearman's correlation coefficients suggest weak positive monotonic correlations between the activity *A1* and the statements *I1* (r_s=.20, p<.05), *I2* (r_s=.26, p<.01), *I3* (r_s=.29, p<.01), and *I4* (r_s=.27, p<.01). On the other hand, the more often participants create programs from scratch, the less they consider the lack of intuitiveness (*I3*) and the risk of getting undesired results (*I4*) as impediments. The Spearman's correlation coefficients indicate weak negative correlations between the activity *A4* and statements *I3* (r_s=-0.22, p<.01) and *I4* (r_s=-0.28, p<.01). Figure 3 illustrates the mentioned relations between the pairs *I3-A3* and *I3-A4*.

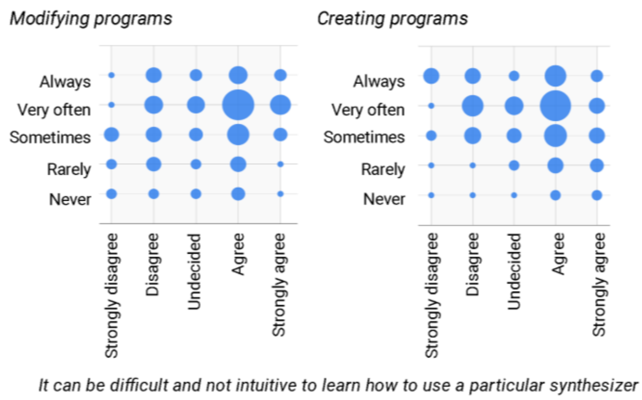

Figure 3. Left: relation between the statement about difficulty of synthesizer programming and the frequency of modifying existing programs. Right: relation between the statement about difficulty and the frequency of creating programs from scratch.

The participants could optionally answer an open-ended question to expand on other challenges they face when creating and modifying programs manually. Out of 122 participants, 34 of them decided to take the opportunity and share their opinion. Most of the answers can be organized in four main groups: 1) challenges related to user interfaces (11 answers), 2) challenges related to learning and understanding the synthesis process (7 answers), 3) challenges related to limited or missing features of specific sound synthesizers (7 answers), and 4) challenges rooted in the creative process (5 answers).

Inefficient user interfaces were most frequently mentioned in the participants' comments. Some of them focused on problems with deep menus (e.g. "*Straightfoward vs menu-divey interfaces*" and "*Menu diving. Wish more manufacturers would surface more of their controls.*"), while the others criticized inconsistency (e.g. "*Some knobs are named different for the same effect*" and "*Thinking more of VI synths - there is so much inconsistency in the UI design that much time is lost understanding what the devs actually want you to do. In contrast, physical synths often (though certainly not always) offered a clearer view of the signal path, simply by their physical layout.*").

The participants also identified a lot of challenges related to learning, especially due to diversity among sound synthesizers (e.g. "*All synths are so different in character, knobs, etc, it takes time to get used to them*" and "*Different sorts of synthesis require different background* *knowledge, most of which have steep learning curves that are at least partially exclusive. In other words, there is an enormous investment of time to deeply learn how the different forms of synthesis work. This learning is a prerequisite to effective use of synthesizers.*”).

A significant number of comments touched upon limitations and lacking features in sound synthesizers. Some examples are: “*Running into 'dead ends', i.e. discovering that a certain function or effect is needed to achieve the desired result, e.g. delay or an extra LFO, or settings not reaching far enough.*”, “*Usually only limitations of that synth, or polyphonic Vs monophonic, number of oscillators*”, and “*The limitation of the synthsiser, in that they all have their own "sound" (as generally defined by it's Oscilators and Filters) and so if you're aiming for a result on the edge of that "sound" then you can get close to or (worse) hit the limits of that synth*”.

Finally, the comments about the creative process were not directly related to sound synthesizers, but opened interesting concerns highly relevant in the context of synthesizer programming, e.g.: “*Building the acoustic landscape across multiple patches*”, “*Having a listen to inspirations and being 100% unmotivated to even try*”, and “*If working with other musicians, who aren't present, your sound cannot be considered complete until you've played it in context with the other parts*”.

4.4 Facilitating Synthesizer Programming

The last part of the questionnaire focused on aspects that help users in creating and modifying programs manually. The first question in that section had a categorical list of all improvements from which the participants could choose exactly one that could help them most, or write their own answer in the “Other” category. The participants mainly opted for intuitive user interfaces (58.1%), informative guides on how to use the synthesizer such as manuals, tutorials, and online material (25.8%), and excellent presets that can inspire users or serve as a starting point for modification (11.3%).

The participants who selected one of those three most frequent answers have been divided in three groups based on their answers. Usage habits between those groups were compared using a set of Wilcoxon rank-sum tests. The results indicate that the participants who think that the user interface can help them most create new programs more often ($p<.01$), but modify existing programs less often than the participants who think that excellent presets can help them most ($p<.05$). Other statistically significant differences have not been found.

In the last question, the participants were asked to rate potential helpfulness of the following functions in creating synthesizer programs: 1) the user chooses a category and the system generates new, random programs that fit the category (*F1*), 2) The user describes a desired sound using attributes (e.g. bright and percussive) and the system generates such a program (*F2*), 3) the user provides an audio sample and the system generates a program that sounds similarly (*F3*), and 4) the user manipulates the graphical interpretation of the sound using an intuitive GUI and the system modifies the program appropriately (*F4*). The results of the Wilcoxon rank-sum tests between all pairs of activities indicate that the participants consider functions *F3* (Median=*Helpful*) and *F4* (Median=*Helpful*) more helpful than functions *F1* (Median=*Slightly helpful*) and *F2* (Median=*Slightly helpful*). All the Wilcoxon rank-sum tests conducted between the following pairs: *F1-F3*, *F1-F4*, *F2-F3*, *F2-F4* confirmed statistically significant differences between the answers ($p<.05$ for all the aforementioned pairs), while no such differences were found between *F1-F2* and *F3-F4*. These findings have been made considering all participants, but the same results have been obtained for the participants without formal music education and those who are not professional musicians or music students.

The participants with a higher music education might consider using attributes (*F2*) more helpful, as indicated by a weak positive Spearman’s correlation coefficient ($r_s=0.23$, $p<.05$). No monotonic correlation has been found between the participants’ usage experience and the helpfulness of the proposed functions. Still, usage habits seem to be related to the perception of helpfulness: the more often participants modify existing programs, the more helpful they consider all the functions (Spearman’s correlation for *F1*: $r_s=0.24$, $p<.01$, for *F2*: $r_s=0.22$, $p<.05$, for *F3*: $r_s=0.29$, $p<.01$, and for *F4*: $r_s=0.15$, $p<.01$). On the other hand, the more often the participants create new programs from scratch, the less helpful they consider function *F2* (Spearman’s correlation: $r_s=-0.22$, $p<.05$).

5. DISCUSSION

Before discussing the results, this section starts with several topics regarding the methodology and scope that are important for interpreting the results.

Since the study primarily relies on the quantitative survey methodology, some known potential biases can affect the results. The questionnaire was carefully designed to minimize those biases: the questions were formulated showing a neutral stance toward different answers, while the terminology was selected and refined during test runs to be as accurate as possible. Also, the introductory text emphasized that the goal of this independent research is to better understand attitudes and habits regarding programming sound synthesizers. However, one particular type of biases, which could have appeared in this research, was not fully controllable by the survey design and the selection of participants. It is the social desirability bias. The questionnaire was disseminated in multiple groups on social networks and online forums that gather synthesizer enthusiasts, hobbyists and professional practitioners, and even researchers in the field of computer music technology. Deep exploration of sound synthesizers, manual synthesizer programming and tweaking synthesis parameters are probably considered as highly respected activities within some of those groups. Even though there is no clear evidence that this fact affected the questionnaire results, the social desirability bias and post-rationalization represent possible risks for quantitative data regarding usage habits. To explore these risks further and mitigate them in future studies, different research methodologies can be used such as diary/camera studies or unmoderated user experience studies.

Another important observation is that habits may be related with purposes of using sound synthesizers (e.g. studio recording vs. live performance, playing different instruments vs. experimenting with sounds, different music genres, etc.). Additionally, as mentioned by one of the participants in the Facebook comments, usage habits may be different for different types of sound synthesizers, especially because of differences between hardware and software synthesizers that might have various concepts of user interfaces and incomparable levels of affordability. Although the questionnaire was designed to cover multiple dimensions, the questions about usage purposes and types of sound synthesizers were not included. The reason is that those topics would require multiple additional questions, as participants may use various types of synthesizers for various purposes. Such extensions of the questionnaire would significantly increase the complexity of analysis and broaden the scope of the study, possibly removing the focus from the current research questions. However, that does not mean that purposes and types are not important dimensions. Understanding users' habits and attitudes in relation to purposes and types of sound synthesizers may be very valuable insights for making comprehensive conclusions. Now, when this study has shown that the music education and the usage experience do not have a significant impact on the usage habits, future research can be more focused on purposes and synthesizer types.

One of the most notable finding in this research is the fact that the questionnaire participants more often modify or create programs manually than they use presets or programs created by others. This is especially interesting because apparently such habits do not depend on music education or experience in using sound synthesizer. A possible concern is that this conclusion may be specific to the group of participants involved in this study. If the group contained more keyboard players who prefer using imitative sounds, the percentage of participants who often rely on existing programs would probably be higher. However, since there were 122 participants acquired from multiple online forums and Facebook groups, the group size and the acquisition procedure should have mitigated a potentially strong sampling bias. Considering the number of participants and their experience in using sound synthesizers, it is valuable to quantitatively analyze habits and attitudes habits of such users, and the results are at least indicative. The synthesizer users similar to the survey participants seem to enjoy the process of creating novel and authentic sounds. The less experienced participants more often acknowledge the fear of getting undesired results by manual synthesizer programming, but that does not seem to demotivate them, as their habits are same as the habits of more experienced users.

Another consistent conclusion is related to user interfaces. A significant number of the participants stated that better user interfaces could help them most in synthesizer programming. They also mentioned various specific problems with user interfaces within their open-ended responses. Knowing that the users generally modify or create programs quite often, user interfaces are inevitably the crucial medium between the user and the synthesis engine, strongly influencing the perception, expectations, and general experience with synthesizer programming. This seems to be recognized by manufacturers of hardware and developers of software synthesizers, as layouts with lots of direct controllers have restored their popularity during the last decade. Another recent trend are hardware devices – so called synthesizer programmers – that can be attached to synthesizers in order to extend their user interfaces. Together with the results of this study, the recent trends provide evidence about the importance of user interfaces. For that reason, user research practices should have a very high priority among those research activities aimed at improving user experience with synthesizer programming. All pragmatically-oriented and technical solutions should be grounded on the UX studies, but this is not the case at the moment.

The open-ended response revealed one interesting point that was not covered by predefined questions in the survey. Some of the participants mentioned limitations or missing features in sound synthesizers as a problem that reflects on synthesizer programming. Therefore, it seems that improving user interfaces may not be sufficient to improve the general user experience. Evidently, some users feel that they sometimes cannot achieve desired sounds, not because of inefficient user interfaces or their lack of knowledge, but because of characteristics of underlying synthesis engines. User interfaces together with synthesis engines have an inseparable effect on sound creation, so both parts should be designed by following informed choices based on UX research.

This study has also shown what the participants think about functionalities for automatic selection of synthesis parameters. The corresponding question was deliberately formed to cover the main approaches explored the previous work in this field. Since the participants expressed more hope in potential helpfulness of target sound matching and GUI-based methods, the results should be interpreted carefully, as the participants did not have an opportunity to try those functions in practice or learn more about them, so they could have had very different ideas about the mentioned functions. For that reason, the future research direction should not be based on this single question, but the results are again indicative, especially the fact that the participants, who modify programs more often, consider all of those functions more potentially helpful. It is generally encouraging to see a positive or at least neutral attitude toward such novel and non-standard approaches.

With other results taken into consideration, it seems that supportive technology should only partially facilitate synthesizer programming, and not fully take control. The participants like to modify and create programs, and technology should help and inspire them, not hinder their creative engagement. In practical sense, that would mean introducing more interactive possibilities [15-17, 31] or generating multiple programs that users can selectively apply for further modifications. The latter concept can be inherently supported by all of those algorithms that rank potential programs and then present only the best one as a result. Examples are solutions based on genetic algorithms that can be easily extended to present multiple programs to users.

6. CONCLUSION

The conclusions outlined in this section are based on the quantitative results and the subsequent discussion, except the first one that emerges from the literature review. The presented conclusions can serve as inputs for synthesizer design, future studies on automatic selection of synthesis parameters, and future user research in the field of sound synthesis.

The first conclusion, based on the literature review, concerns the observed lack of user research in the existing solutions for automatic selection of synthesis parameters. While technical solutions employ advanced computer science techniques to resolve the problem of synthesizer programming, there is no evidence that the problem is appropriately formulated. Some of the previous studies conducted user testing, but only to demonstrate that solutions work well. The missing part is an investigation whether the solution would be more usable if it was based on a different approach. User experience studies should serve as one of information sources when deciding upon the solution's architecture and its argumentation in scientific publications.

The second conclusion is one of the most important quantitative results of this study. It is the fact that the participants more often modify or create programs manually than the use existing presets and programs. This result can influence the future direction of developing solutions for automatic parameter selection that target users similar to the participants of this research. Instead of aiming at synthesizing final sounds, those solutions could be designed to efficiently support the synthesizer programming process that enthusiastic users apparently prefer over using existing programs.

The existing and missing correlations in the result suggest that the habits regarding synthesizer programming are not related to user's music education or experience, but on the other hand, they are related to users' perception of impediments and helpfulness of possible solutions. For example, the participants who modify existing programs more often, agreed more with all the impediments and also considered all proposed functions more potentially helpful, but that was not the case with the users who create programs more often. Of course, the correlations do not confirm causalities and it is not possible to conclude whether habits form a perception of impediments, impediments form habits, or those dimensions are not causally related at all. However, the correlations are a very important reminder that not all users are the same and that particular solutions should aim to satisfy specific needs and expectations. When designing a novel solution for automating selection of synthesis parameters, a starting point should be based on the intended purpose and target users.

Finally, as a general remark regarding possible solutions for more efficient synthesizer programming, the results of this study show that the participants believe that the most helpful improvements would be those in user interfaces. While this result may be affected by the fact that users perceive the sound synthesis technology and its possibilities through user interfaces and thereby assign all problems and potential solutions to the interface level, this is still an interesting insight, especially for practically-oriented solutions. Improvements of user interfaces or interactive approaches to automatic synthesis parameters selection are surely not the only mean of facilitating synthesizer programming, but they may be a safe starting point. Although the results of this research may provide some general guidelines for user interface design and overall solution conceptualization, they are not sufficient to inform all design decisions, as their purpose was to provide insights in habits and attitudes regarding synthesis programming and not to answer specific questions. Therefore, the design process should be informed by a carefully conducted user research based on the appropriate methodology.

7. REFERENCES

[1] E. Miranda, "An artificial intelligence approach to sound design," in Computer Music Journal, vol. 19, no. 2, 1995, pp. 59–75.

[2] A. Seago, S, Holland, and P. Mulholland, "A Critical analysis of synthesizer user interfaces for timbre," in Proceedings of the XVIII British HCI Group Annual Conference, UK, 2004.

[3] C. Rasmussen, "Evaluating the Usability of Software Synthesizers: An Analysis and First Approach", MSc Thesis, University of Guelph, 2018

[4] A. Horner, J. Beauchamp, and L. Haken, "Machine tongues XVI: Genetic algorithms and their application to FM matching synthesis," in Computer Music Journal, Vol. 17, No. 4, 1993, pp. 17–29.

[5] A. Horner, "Wavetable Matching Synthesis of Dynamic Instruments with Genetic Algorithms," in Journal of the Audio Engineering Society, Vol. 43, No. 11, 1995, pp. 916–931.

[6] A. Horner, "Auto-Programmable FM and Wavetable Synthesizers," in Contemporary Music Review, Vol. 22, No. 3, 2003, p. 21–29.

[7] S. Wun and A. Horner, "A Comparison Between Local Search and Genetic Algorithm Methods for Wavetable Matching," in Journal of the Audio Engineering Society, Vol. 53, No. 4, 2005, pp. 314–325.

[8] J. Riionheimo and V. Välimäki, "Parameter Estimation of a Plucked String Synthesis Model Using a Genetic Algorithm with Perceptual Fitness Calculation," in EURASIP Journal on Applied Signal Processing, Vol. 8, 2003, pp. 791–805.

[9] M. J. Yee-King and M. Roth, "SynthBot – An Unsupervised Software Synthesizer Programmer," in Proceedings of the International Computer Music Conference, Ireland, 2008.

[10] J. McDermott, "Evolutionary Computation Applied to the Control of Sound Synthesis", PhD Thesis, University of Limerick, Ireland, 2008.

[11] A. Horner, “Envelope Matching with Genetic Algorithms,” Journal of New Music Research, Vol. 24, No. 4, 1995, pp. 318–341.

[12] M. J. Yee-King, “The Evolving Drum Machine”, Music-AL workshop, ECAL Conference, 2007

[13] M. Chinen and N. Osaka, “Genesynth: Noise Band-Based Genetic Algorithm Analysis/Synthesis Framework,” in Proceedings of the International Computer Music Conference, Denmark, 2007

[14] I. Fujinaga and J. Vantomme, “Genetic Algorithms as a Method for Granular Synthesis Regulation,” in Proceedings of the International Computer Music Conference, Denmark, 1994, pp. 138–138.

[15] J. Young, “Rethinking Synthesis: Extending and Exploring Gendyn,” BA Thesis, University of Sussex, UK, 2010

[16] M. Macret, “Automatic tuning of the OP-1 synthesizer using a multi-objective genetic algorithm,” in Proceedings of the Sound and Music Conference, 2013, pp. 614–621.

[17] C. Johnson, “Exploring Sound-Space with Interactive Genetic Algorithms,” in Leonardo, Vol. 36, No. 1, 2003, pp. 51–54.

[18] P. Dahlstedt, “A MutaSynth in Parameter Space: Interactive Composition through Evolution”, Organised Sound, Vol. 6, No. 2, 2001, pp. 121–124.

[19] J. Mandelis, and P. Husbands, “Musical Interaction with Artificial Life Forms: Sound Synthesis and Performance Mappings,” in Contemporary Music Review, Vol. 22, No. 3, 2003, pp. 69–77.

[20] B. Hamandicharef and E. Ifeachor, “Intelligent and Perceptual-Based Approach to Musical Instruments Sound Design,” in Expert Systems and Applications, Vol. 39, No. 7, 2012, pp 6476–6484.

[21] M. J. Yee-King, L. Fedden, and M. d'Inverno, “Automatic Programming of VST Sound Synthesizers Using Deep Networks and Other Techniques,” in IEEE Transactions on Emerging Topics in Computational Intelligence, Vol. 2, No. 2, 2018, pp. 150–159.

[22] D. Wessel, “Timbre Space as a Musical Control Structure,” in Computer Music Journal, Vol. 3, No. 2, 1979, pp. 45–52.

[23] C. Hourdin, G. Charbonneau, and T. Moussa, “A Sound Synthesis Technique Based on Multidimensional Scaling of Spectra,” in Computer Music Journal, Vol. 21, No. 2, 1997, pp. 40–55.

[24] T. Jehan, “Perceptual Synthesis Engine: An Audio-Driven Timbre Generator,” MSc Thesis, Massachusetts Institute of Technology, 2001.

[25] J. W. Beauchamp, The Sound of Music: Analysis, Synthesis, and Perception of Musical Sounds, Springer, New York, 2007.

[26] D. Mintz, “Towards Timbral Synthesis: A New Method for Synthesizing Sound Based on Timbre Description Schemes,” Master thesis, University of California, Santa Barbara, 2007.

[27] A. Gounaropoulos and C. Johnson, “Synthesising Timbres and Timbre Changes from Adjectives/Adverbs,” in P. Collet et al. Applications of Evolutionary Computing, Springer-Verlag, Berlin, 2006.

[28] G. Kreković, A. Pošćić, D. Petrinović, “An Algorithm for Controlling Arbitrary Sound Synthesizers Using Adjectives,” in Journal of New Music Research, Vol. 4, No. 45, 2016, pp. 375–390.

[29] R. Ashley, “A Knowledge-Based Approach to Assistance in Timbral Design,” in Proceedings of the International Computer Music Conference, The Hague, Netherlands, 1986.

[30] R. Ethington and B. Punch, “SeaWave: A System for Musical Timbre Description,” in Computer Music Journal, Vol. 18, No. 1, 1994, pp. 30–39.

[31] R. Clement, “Automatic Synthesiser Programming,” in Proceedings of the International Computer Music Conference, Universitiy of Huddersfield, UK, 2011, pp. 155–158

[32] A. Pošćić and G. Kreković, “Controlling a Sound Synthesizer Using Timbral Attributes,” in Proceedings of the Sound and Music Computing Conference, Stockholm, Sweden, 2013, pp. 467–472.

[33] G. Kreković and D. Petrinović, “Intelligent Exploration of Sound Spaces Using Decision Trees and Evolutionary Approach,” in Proceedings of the International Computer Music Conference joint with Sound and Music Computing Conference, Athens, Greece, 2014, pp. 1263–1270.

[34] A. Pošćić, G. Kreković, and A. Butković, “Desirable Aspects of Visual Programming Languages for Different Applications in Music Creation,” in Proc. Int. Conf. Sound and Music Computing (SMC2015), Maynooth, 2015, pp. 329-336.

[35] A. Pošćić and G. Kreković, “The Frailty of Formal Education: Visual Paradigms and Music Creation,” in Proceedings of the Audio Mostly, London, 2017

[36] A. Pošćić and G. Kreković, “Ecosystems of Visual Programming Languages for Music Creation: A Quantitative Study”, in Journal of the Audio Engineering Society, Vol. 66, No. 6, 2018, pp. 486-494.

[37] V. Armstrong, "Hard bargaining on the hard drive: gender bias in the music technology classroom," Gender and Education, Vol. 20, No. 4, 2008