ELSE: AN ARTISTIC STUDY ON ALGORITHMIC AGENCY IN SOUND SYNTHESIS COMPOSITION

Daniele Pozzi
Computer Music Project - WS 2019
Supervision: David Pirrò

ABSTRACT

This article presents the development of a SuperCollider patch as a case study of the influence of algorithmic processes and algorithmic thinking in computer music practice. The crucial stages involved in the experimental design of a sound synthesis algorithm are analysed from a critical perspective that aims at exploring the multifaceted dynamics underpinning the relationship between artist and algorithmic process. In doing this, I'll try to highlight the different agencies that contribute to shape the creative process of composing music with computers. It is suggested that the process of constant reconfiguration, intended as the iterative adaptation that takes place between an artist and the algorithms employed, has a major generative role in algorithmic practice. We may identify this generative element as one specific expression of the agency of the algorithmic. The case study is based on a retrospective documentation of *Else*, a generative algorithm conceived and realised in SuperCollider.

1. INTRODUCTION

Looking back at the history of music we may notice that each new technological development has provided musical composition with an altered access to sonic reality [3]. Literature is rich in examples of tools and techniques that radically modified our relationship to music by opening new unforeseen possibilities: record players and domestic listening, or tape recorders and acousmatic music are well-known examples of technical objects that fostered new ways of composing, perceiving and thinking of music. The idea of exploring the possible musical relationships that emerge by engaging with certain techniques or tools is notably fundamental to electronic music. German-Dutch composer Gottfried Michael Koenig writes that, when composing electronic pieces, he has "always searched for causes in the technical conditions of the studio. [...] the machines should not only be used economically, but also musically, they should take over form building tasks [4]". Indeed, one of the peculiar traits of electronic music is that its means of production transcend their instrumental function, becoming an active agency, a generative element in the compositional process.

Along this line of thinking, this article focuses on computer music practice and its major means of expression: algorithms. Algorithms and their active agency are the object of this study that centers around the development of *Else*, a generative system composed in the SuperCollider framework. Based on constant and reiterated stages of self-observation and consequent re-adaptation, this process can be understood in the context of artistic research; every decision has been taken after the critical evaluation of the outcomes of the previous stages of aesthetic experimentation, leading to various intermediate stages of development that we will try to dissect and analyze in the next sections.

1.1 Algorithms

In the context of computer science and mathematics the most common definition of algorithms could be summarized as the abstract representation of a set of rules needed for solving a problem. Algorithms are thus considered in terms of a functional understanding: they exist in order to address problems. They are abstract, immaterial and embody a human thinking process.

I think that this interpretation of algorithms as functional projections of our mental abilities falls short in capturing the specificity of the algorithmic medium, especially in the context of contemporary technological developments and their uses, which demand for a shift in this definition. This deviation is recently stressed by various artists and scholars in transversal fields such as software studies, cultural theory and software art. A common thread is the individuation of an *incompressible* quality of the *algorithmic* [9], which may be found in their performative dimension or in an inherent agency. For instance, artist and researcher Hanns Holger Rutz writes:

> Although algorithms have inspired the electronic art for a long time, the understanding of their aesthetic consequences has changed over time and currently is undergoing another shift, possibly the reason for the renewed interest in their artistic use. [...] Algorithms now bear a crucial relationship to material reality, they can have unintended consequences, they can crash machines, etc. Algorithms have become *performing entities*. [6]

Far from being the passive recipients of our cognitive functions, algorithms exhibit an active agency which can be witnessed in many situations. Providing algorithms with agency has profound aesthetic consequences in artistic practice. Apart from a few current research in computer music, e.g. the Algorithms That Matter project [7], implications of this different understanding are not yet understood and need to be further investigated and explored. What is the generative potential of algorithms in computer music? What is the peculiar *altered access* they provide to sonic reality? How do they actively shape and remediate our understanding of music? Which new modes of perception, interaction and composition do they afford? These questions drove the research process I present in this paper, whose main sections are structured as follows: Section 2 provides a survey of the concept of algorithmic agency. Section 3 describes the process of development of *Else*, a software artifact whose development was driven by an aesthetic exploration of the dynamics implied in algorithmic agency. Section 4 summarise the different threads that this artistic process revealed.

2. ALGORITHMIC AGENCY

In this section I'll try to clarify my understanding of the concept of *algorithmic agency* and possible ways to explore it. I relate the term to the performative dimension of algorithms, to their potential of shaping and remediating the practice of computer music. Departing from a subjective analysis of the influence exerted by the algorithmic medium on compositional activity and its artifacts I iteratively adapted and deviated my practice, tracing a path that investigates and questions the peculiarities of composing music with computers.

In order to achieve this tight intertwinement between artist and algorithm, I adopted an ecological perspective that tries to counterbalance the more functional approach, that treats computation as a mean to achieve a pre determined result, prioritizing the composer's will over its implementation. A key point in our understanding is the idea that these elements - the aesthetic desire and its realization - are not intended as divided and sequentially or functionally connected: they are rather interdependent mechanisms that actively interact and adapt each other, at the same time defining the artistic process and its development.

This approach led to an *abstract* form of sound synthesis, one which is not derived by a super-ordinated model but rather evolves accordingly to subjective compositional ideas of sound and musical organization. It may be understood as a form of *compositionally motivated sound synthesis* [3] or, as defined by Holtzman, as a non-standard synthesis method that "given a set of instructions, relates them one to another in terms of a system which makes no reference to some super-ordinated model, [...] and the relationships formed are themselves the description of the sound [10]". The term *non-standard* is used in opposition to the *standard* synthesis methods, which are traditionally formulated after physical, acoustical of psychoacoustical phenomena and evaluated on the basis of their intended perceptual appearance. From the non-standard standpoint sound is rather understood in terms of the process that constructs it, and "the computer acts as a sound-generating instrument *sui generis*, not imitating mechanical instruments or theoretical acoustic models [5]".

My intuition is that this approach facilitates an ecological relation between artist and algorithm, fostering an investigation of algorithmic agency and the dynamics of computer music practice. In order to have a better look at these themes a profound engagement is required, which cannot be achieved by separating the composer from his or her processes. I rather think that algorithmic agency is to be found at their *intersection*, and that it could be better understood through a meticulous examination of the network of interdependencies that develops out of this relation.

The survey of these themes and their actual implementation led to the development of *Else*. Many themes emerged along the path of its development, and the next sections attempt a retrospective description of the decisions which were taken, together with their practical and conceptual implications.

3. ELSE

Else is the arriving point of a process of iterative reformulation of a SuperCollider patch, whose algorithm evolved according to the new aesthetic directions and desires emerging out of the direct interaction with the program and its sound outcomes. Starting from a quite traditional instrument based on frequency modulation synthesis, *else* went through diverse development stages: at first it followed a path of incremental complication, passing through the introduction of recursive feedback processes, eventually coming to an iterative unpacking and reformulation of its same structure, following a *reductive* approach. For sake of clarity, these different steps are here presented as distinct and sequential, even though in the actual praxis they were sometimes combined or alternated. Nevertheless, all the strategies described in this section shared a common aesthetic and compositional aim, that of expanding the generative capabilities of the system itself, looking for a way of describing an algorithmic process whose sound outcomes could escape its same formulation.

3.1 Original Patch

The original patch was first employed in the context of an improvisative electroacoustic duo - viola and laptop - and its design was based on a spectral approach to digital synthesis for the rapid prototyping of diverse timbres. It was essentially composed of four synthesis units all sharing the same design, each of which was operating in a different frequency range of the audible spectrum (low, mid-low, mid-high, high). Each unit comprised three modules, or sub-units: a synthesis component, consisting of a SuperCollider Ugen called FM7, essentially emulating the Yamaha DX7 synthesizer - six sinusoidal oscillators could be freely combined to create nested frequency modulation structures; an effect module, mainly composed of different reverbs and compressors, and an EQ module at the tail of the synth graph completed the instrument structure.

3.2 Complication

Although this design favored the quick sketch of different timbres, these were mostly static and the patch was not capable of generating much variety. A first strategy to contrast this lack of temporal variation was to introduce modulating signals, especially to alter the equilibrium of the FM7 plugin. The behavior of this unit is described by defining two matrices, in the exact same way one would do on a real Yamaha DX7: the first one controls the frequency, phase and amplitude of each of the six oscillators, the second the amount of modulation each oscillator's output has on another oscillator's phase. These matrices are usually filled up with static variables, especially when trying to emulate the DX7 sonorities. By adding a further modulation layer on top of these values the patch gradually began to generate more dynamic timbres, with the time-varying modulation indexes modifying the harmonic content and the broadness of the spectrum.

This approach was extended to other parts of the algorithm, especially compressors and filters, by iteratively adding new components - mainly sine, square and saw modulators - on top of the pre-existing ones. The result was a complicated network of intermodulations, counting a large number of SuperCollider UGens arranged in a nested structure. Nevertheless, it was possible to minutely sculpt the overall sound by directly addressing the individual components: many fine-tuning adjustments were viable and the overall program was well controllable in an instrumental fashion. This relatively large digital instrument turned well suitable in the context of the duo both when rehearsing and playing, where it was necessary to have a synthesis patch that could be played through a linear and responsive interaction.

3.3 Otherness

In parallel to this duo scenario, the desire emerged to transpose the instrument in a live-electronics solo setting. Some minor changes were made for it to be played through a midi box, using faders and knobs to improvise by modifying different parameters of the patch. Upon some first experiments, it was evident that something was missing: the way the instrument was conceived proved to be insufficient to fill the gap created by the absence of the second musician. Indeed, the lack of this other external, unpredictable stimulus to confront with unraveled some characteristics of the instrument itself, which previously went unnoticed. The excess of controls available and the rigid hierarchical configuration of modulators were favoring a stiff relationship with the algorithm: the interaction was linear and the result easy to predict, transforming the act of playing in a mere execution, which had to be planned ahead because of the amount of actions needed to produce meaningful temporal developments.

At this point a radical change was needed, and a new question emerged: how could such system be modified in order to regain this crucial presence of *otherness*? Which dynamics had to be altered to confront again with an unpredictable situation, with something able to create surprise, something which could enable different types of - unexpected - engagements?

My reaction was that of further complicating the patch, expanding its functionalities through the reiterated addition of UGens, hoping that an extreme complication could reproduce this sense of surprise by making the overall dynamics of the system more obscure. Soon this process turned in a tedious, endless search for the perfect coverage of the algorithm itself. But actually, the program was just getting bulkier, more devised and more rigid.

3.4 Feedback

I understood that a different kind of algorithmic intervention was needed, something that could radically alter the internal dynamics of the patch. Upon noting that the rigidity I was experiencing could be attributed to the hierarchical structure of the program, I decided to abandon the complication strategy based on nested modulation. I introduced internal feedback in the patch, achieving a circular organization in which the output of each component could influence the state of any other. By working at the level of the *relationships* between the elements of the algorithm the system slowly began to behave differently, producing less trivial results. Because of the feedback, the interactions happening between the different components were no more linear and sudden and larger changes could now happen between each cycle.

This non linearity was also reflected at the level of the interaction with the algorithm. If the consequence of modifying a parameter was previously easy to predict, this same action could now lead to surprising, unexpected results. Due to this new organization, the patch began to loose its functional, instrumental character, slowly drifting towards a more independent system that could partially evoke this sense of *otherness* I was after.

3.5 Identity

But how could this sense of otherness be pushed forward? All these sudden changes happening in the system, all the extreme jumps between different sounds were still taking place within the boundary of a well defined timbral space. It was virtually impossible to cross this boundary, to *exceed* the space of possibilities defined by the rigid combination of the UGens that composed the system. This precise border, this line that marked what could happen and what could not, lend a strong sense of identity to the algorithm. This emerging identity corresponded to the projection of my aesthetic decisions at the time of composing the system. By deciding which UGens could be part of the algorithm and which couldn't I was constructing a well defined space of possibilities, a virtual border between the system and everything else. This projection of my personal aesthetic beliefs, this *imposed* identity was somehow contrasting with the search for the otherness. Since I found a way to model the interactions between the components so that these could escape my projections, could this be done also at the level of the components themselves? Is it possible to model an algorithmic system capable of *generating* its own components? How can an algorithm produce its own identity?

3.6 Subtraction

These are obviously utopian questions, even though related problems are addressed by some branches of computer science such as evolutionary [11] or autopoietic computing [12, 13]. Still we know that algorithms need to be written in the form of "well defined set of instructions". Nevertheless in the realm of sound algorithms we can investigate the concept of identity from other, more tangential, perspectives. It's not unusual for a synthesis patch to have some components showing a stronger influence than others on the type of sounds which are produced. We may say: "Sounds like granular synthesis" or "This sound is quite FMish" to describe processes which are often more complex: the elements we individuate are usually part of larger synthesis systems. Nevertheless our attention is drawn to specific components which, depending on the organization of the algorithm, happen to have perceptual predominance, strongly influencing the sound identity of an algorithm. For example, I noticed that in my patch some effect units like reverbs or filters were leaving a clear trace of their function on the overall timbre, shadowing part of the dynamics of the overall system. I thus decided to subtract these components from the algorithm, and to continue working exclusively on the synthesis using basic sound generators.

3.7 Specificity

Trying to further dig in the issue of the identity of the algorithm, I started questioning which specific dynamics brought the patch to its actual form: how did I get to this exact set of UGens I was using? As said before, this corresponded to the projection of reiterated aesthetic decisions in the form of a sound algorithm, in which each new addition or alteration of the graph structure emerged out of the process of experimentation with the system. Every change represented an attempt to modify the patch so as to steer its output towards a particular sound result I was imagining. This practice was mainly happening through the manipulation of SuperCollider Ugens: high-level sound objects - like the FM7 plugin or different kinds of oscillators - having a specific behavior, which could thus convey a specific aesthetic intention. I had the feeling that these aesthetic intentions played a major role in constructing the identity of the patch, thus I decided to try to experiment with this particular aspect of sound synthesis composition: if the usage of high level, specific operators was the medium through which I could express my desires in the realm of sound synthesis, why don't I try to reconsider this same system, reformulating it by using other less specific operators? I thought this would be an effective strategy in preserving the overall aesthetic direction of the algorithm, while at the same time trying to counter balance the importance of my personal desires in the dynamics that contributed to shape its identity. Through this act of reformulation my intentions would be partially shadowed, potentially originating a different kind of algorithm, whose identity would be less dependent on my imposed desires and more emergent out of the actual praxis of reformulation.

3.8 Unpacking

I began by individuating specific parts of the algorithm: small mechanisms composed of high level UGens which together fulfilled a well defined function within the overall structure of the program. I then proceeded in *unpacking* these functionalities to gain an access to processes which were previously encapsulated in a pre-given relationship. Such mechanisms were then reformulated by means of lower-level operators, making the original process available to a recomposition and a reconnection. Less specific aggregates of objects were thus created, expanding the space of possible results and transgressing the borders of their original function.

It is useful to present a practical example of unpacking and reformulation. One subsystem I individuated was composed of two SuperCollider UGens: a zero crossing detector and a sine oscillator. This compound had the purpose of analyzing an internal signal and generate a sinusoidal waveform that would then modulate other components of the patch. The frequency of this sinusoid was directly derived by the pitch estimation performed by the zero crossing.

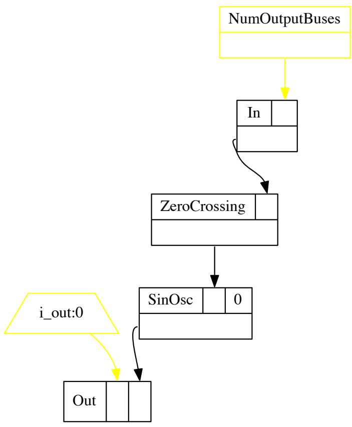

Figure 1. Mechanism before reformulation.

The unit was reformulated using a BufWr and BufRd object operating on a shared audio buffer - Figure 2. These components respectively write and read data in and from the buffer, at a specific index which is provided externally (usually by a phasor). The BufWr is used to create a ring buffer: an input phasor (1) iterates from 0 to buffer lenght. The BufRd behavior is dependent on the sign of the input: every time the signal crosses the zero, jumping from a negative to positive value, the actual index of the writing phasor is sampled (2). This index represents the end of the ramp of another reading phasor (4), whose start value is the index sampled at the previous cycle (3). The remaining code concerns the wrap around function for the reading phasor - actually, two parallel reading phasors are used to implement the wrap around function.

This reformulation is a good example of how to exploit some elements which are already present in the signal path in order to *approximate* a specific functionality, while at the same time expanding the possible outcomes of a specific mechanism. Before its reformulation, the original subunit could only generate a very specific kind of result: a sinusoidal signal. This was due to the use of a specialized UGen like SinOsc, which generates a new signal according to a sinusoidal function. In reformulating the whole process I used signals which were already present in the algorithm, removing this specialized operator, thus subtracting its highly specific output which was constraining the possible outcomes of the mechanism exclusively to a sine wave. The new reformulated unit is capable of approximating a sine wave under specific circumstances - especially at higher frequencies, see Figure 3, but it can also produce other less specific types of waveforms. This could be understood as an expansion of the space of possible results of the overall mechanism: in this new configuration, the sine wave is a *possibility* rather than a *certainty*.

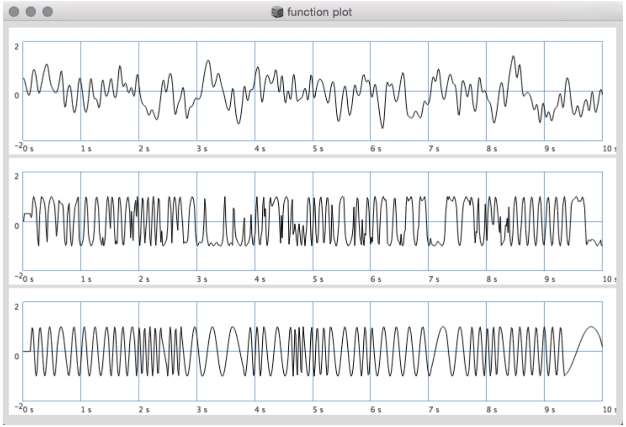

Figure 3. Comparison of the mechanism results before and after reconfiguration. 1st waveform: input signal. 2nd waveform: output after reconfiguration. 3rd waveform: output before reconfiguration.

This type of reformulation aimed at preserving some specific aspects of the output signal of the original unit - its amplitude and frequency - while exceeding its sinusoidal function. The choice of which features are to be maintained and which others can be approximated - or disregarded - is a compositional one. It involves a re-interpretation of the orginal algorithm, evaluating which aspects are important for the ecology of the overall process, with respect to the overall aesthetic direction. The other features - the ”non-fundamental ones” - are then open to experimentation, leading to exploring the generative potential of the process of reformulation. The new sub-system is capable of producing a greater variety of output timbres while being constructed out of simpler building blocks that are now exposed to possible reconfigurations rather than being secluded in a black box.

3.9 Orders of Magnitude

The method of unpacking is in itself a means to investigate the different orders of magnitude one can consider when conceiving a sound synthesis algorithm. It allows us to look at these processes from other angles, stimulating a reflection on the personal representations we use when composing sounds. For example, we commonly understand the ZeroCrossing as an analysis tool: it outputs a certain frequency according to the rate of change of the signal from negative to positive. Thus, we will tend to use this unit in specific contexts where such representation is meaningful or useful. When unpacking and reformulating the mechanism composed of the ZeroCrossing and the SinOsc, the sign detection that was previously encapsulated inside the analysis Ugen became part of a new compound, contributing to the creation of a real-time sampler. This sampler is no more related to the frequency representation which is fundamental to the ZeroCrossing Ugen: the Hertz measure in itself is no more useful at all. The zero crossing detection, which we previously considered only as an analysis tool, suddenly transformed into a trigger element of the new sampler.

Nevertheless, as we have seen in the previous subsection, the output of the new mechanism shares many similarities with the one of the old compound. This mechanism resulted out of the process of unpacking and reformulation: it was formulated by analyzing the old compound at different levels. By considering the mechanism as an whole and *at the same time* looking at its internal components we could create a new object, that was emergent out of the encounter between these two orders of magnitude. I believe I wouldn’t have reached such configuration if I had considered exclusively the high level representation of the mechanism, or if I had used the sign operator without a reference to the ZeroCrossing Ugen.

I employed this method of reformulation to match many mechanisms of the system. Hence, signals have been combined by means of lower-level operations, like addition or multiplication. This process of replacement was conducted following a ”trial and error” methodology, where each operation had to be empirically evaluated before being eventually inserted in the system. This strategy allowed us to directly engage with the sound identity of the algorithm and with my past aesthetic choices: the trajectory I followed was that of expanding or modifying such identity in a direction that could be less dependent on my desires, and identity that was emergent out of the actual process of reformulation.

3.10 Reduction

At first, these reformulations caused an increase in the number of elements in the implementation of the synthesis process, as it is clear when comparing Figure 1 and Figure 2. Since many of those elements appeared as redundant in the system, they could easily be reduced by condensing more into one. But this process of reduction took an even more radical form. By combining other signals already present in the system, and exploiting feedback as a source of sound material, even more reduced formulations of the system could be developed. This reduction was an iterative process, in which, in subsequent steps of condensation, the whole system was observed. This means that at this point, no single element of this synthesis algorithm was considered separately from the whole ecosystem of interconnections it was part of. Each cycle of experimentation produced a reduction of a portion of the system, while preserving its overall behavior.

Thanks to this movement toward abstraction both with respect to its components and the relationships between them, the new algorithm had a more *obscure* identity, producing unpredictable results that contributed in evoking that sense of *otherness* I was after. This system exhibited different performative qualities, therefore demanding for a shift in the way of encountering it. A more interactive relationship was sought that would also enable an explorative rather than instrumental attitude.

But at this point an even more fundamental shift happened: indeed, I began to look at the process of reformulation from a different perspective. If I was previously focused on reformulating specific behaviors while trying to maintain a specific aesthetic output, my interest gradually turned towards the exploration of the new possibilities such method unfolds.

A dedicated GUI was then developed that allowed to rapidly experiment different combinations of signals and internal feedback paths by using a visual matrix in combination with an external midi controller. This setup allowed to experiment with the tight interrelations that define the dynamics of the system, making those immediately perceivable through the variations induced in the sound behavior.

4. THREADS

I might pull out of the experimental process some thread which I think are relevant for their aesthetic and artistic implications. Some of them are collected in this section.

4.1 Complexity and Possibilities

At first the development of *Else* was mainly guided by the desire of expanding the generative capabilities of a synthesis algorithm. A possible way to achieve this expansion is by complicating the system - as described in Section 3.2 - in order to produce a greater variety in its sound results. Through an aesthetic engagement with this kind of process, this method proved to lead to a type of expansion that was qualitatively different from what I intended: I was certainly expanding the sonic capabilities of the system, but still these remained confined in a *predictable* space of possibilities. In other words, the sound results produced by complicating the system could still be easily inferable from the process which generated them, leaving little space for the unintended.

Thus I had to reconsider and redefine what kind of *expansion* I was searching for, and how to reach it. I understood that I searched for an algorithmic process that should have been capable of generating a space of possibilities which is not contained, nor intended in its formulation or implementation. A process that would possess a kind of excess with respect to its writing: I called this quality *complexity*. To reach this quality, in my experimentation I took the path of iterative unpacking and reformulation (as described in Section 3.8 and Section 3.10), that I had the intuition could lead to a radically opposite direction than complication.

4.2 Imperfections and Approximations

A side-effect of this iterative process of reformulation was the introduction of approximations, and therefore slight imperfections and undefined deviations from its initial behavior. Even though these approximations sometimes led to a faulty reconstruction of the original unit, nevertheless they were accepted as being part of the process of reformulation and abandoning of an instrumental approach to algorithms. These deviations can indeed be understood as generative material, and their unintended consequences have a main contribution in increasing the complexity of the system, thus unfolding its potential of exceeding the boundaries of my previous functional approach. Moreover, the removal of specific units could be intended as a reduction of the amount of expectations and intentions contained in the formulation of the system. A direct consequence of this reduction is the generation of aesthetic results which are not intended at the moment of composing the process, nor implicit in the actual code that represents it. This excess which is produced by the algorithmic process causes a shift of the space of possibilities of the system towards the unintended and unpredictable.

4.3 Algorithmic Experimentation

For the sake of clarity, the description of the experimental process was divided into six main cycles (Section 3.1, 3.2, 3.4, 3.6, 3.8, 3.10). It is important to point out that each of these was composed of many sub-iterations, where every algorithmic intervention was informed by the aesthetic result of the previous one. Searching for a form of agency in the algorithmic, I had to proceed step-by-step, evaluating every single change before proceeding to the next. This type of exploration requires a constant and critical involvement in the material practice of composing with algorithms, in the subjective and empirical effort of establishing an alternative mode of relating to the algorithmic medium. I call this iterative experimental process *algorithmic experimentation*.

4.4 Method of Reformulation

I adopted the method of reformulation as a specific form of algorithmic experimentation that, by focusing on removal rather than expansion, forces the composer to reflect on his manual, subjective activity, while at the same time experimenting with the generative potential of algorithms in computer music. This reformulation coincides with the condensation of algorithmic material in favor of creating a less specific system: pre-given algorithmic objects (SC Ugens) which have a specific behavior are unpacked and then rewritten in an approximated version, in terms of new relations between less specific objects already present in the algorithm.

This corresponds to the removal of a specific aesthetic intention which was embodied in that particular behavior. By subtracting this intention I thus obtaine a more balanced distribution of the agencies involved in my relationship with the algorithm. I remove part of my aesthetic wishes in order to make space for something else to happen. This *else* is not pre determined, and cannot be imagined, nor it can exist prior to the direct engagement of the artist with his means of expression. This produces a shift from a functional conception of algorithms to a generative one, where the artist is not an external inventor that acts over materials, but rather an adaptive agent that constantly modifies his desires, methods, aesthetics and artistic vision in the process of directly engaging with the dynamics that underpin artistic activity.

4.5 Productive Engagement

By putting himself in this position, the artist is forced to enter new ways of relating to his materials. A potential effect of this new relation is the production of an alternative aesthetic language, a *deviation* from existing conventions, beliefs and vocabularies. This deviation resonates with Brün's concept of *anticommunication*, which "is an attempt to say something through a channel which is not yet available, not yet established". Brün writes: "communication uses the order and the law that is meant to be recognized by the receiver as the receiver's own; anticommunication creates the order and the law that is meant to be discovered by the receiver for the first time" [1]. Here Brün refers to the receiver as the ultimate recipient of the artwork, but the concept could here be extended to the artist himself that, in his attempt of eschewing his own approach, ends up discovering new ways of relating to his materials that are not grounded in established norms, and new ways of saying something out of this relation.

4.6 Non-standard approach

In many musical contexts predictability and control are desirable, and that's where the functional side of algorithms and computers is really effective. Nevertheless, I think that the functional usage of computers and algorithms prevents the unfolding of their active agency. More precisely, my intuition is that an unbalance between aesthetic intentions and their algorithmic representation might hinder the emergence of the potential effects inherent in the interaction between artist and alogorithm. These effects can only unfold if there is an equilibrium in this relationship and the research process I present here tries to find ways of balancing the weight and dynamics in the compositional ecosystem. The direction I followed was indeed the reduction or even the abandoning of the control implied in the functional approach, therefore searching for alternative accesses to the space of algorithms and leaving room for unintended and unexpected implications.

4.7 Recursion

In this type of approach, a crucial role is played by the recursive cycles of evaluation, modification and re-adaptation of the patch. Since the synthesis target is not defined *a priori*, a constant iterative engagement is required in which each change has to be aesthetically evaluated before proceeding to the next one. A tight feedback between the algorithm and the artist is thus established, and the resulting aesthetic artifact is emergent out of this particular interaction. This kind of recursion is characteristic of the algorithmic medium, since software developing is not an unidirectional activity: it is always possible to go back to a previous state, modify some parts or radically transform the whole program. The algorithmic medium is a *malleable* one, and an algorithm is always available to a recomposition and reconfiguration.

4.8 Threshold of agency

It is interesting to observe how the active engagement in the material practice of experimentation not only fostered novelty in the resulting sonorities, but also induced a shift in terms of new aesthetic desires and modes of relating to the algorithmic medium. These changes can only be tracked back trough a retrospective examination of the overall artistic process, but nonetheless they suggest an active agency of both the medium and process in shaping artistic activity.

Significant shifts were described for example in Section 3.3 and 3.5, when the direct engagement with the algorithmic process actively modified my artistic intentions. At first, actions and methodology were driven by aesthetic expectations and, for a long time, best efforts were made in preserving those sound outcomes which were originally intended, while at the same time slowly enlarging the space of possible results. But, at a specific moment (see Section 3.10, this relationship radically changed: rather than focusing on maintaining a specific sound quality, it became more interesting to explore those possibilities that were emerging out of the process of reformulation. That moment was precisely when the algorithmic process began to *retroact*, altering intentions and methods and modifying the relationship between the computer, the composer and the process itself.

This shift was not previously intended, and emerged out of the process of direct engagement.

4.9 Interaction

I have further experienced that the agency I was searching for doesn't just reside in *Else* or in its generative potential. It is rather to be found in the tight interdependency of all the different actors involved: indeed the actual agency originates from the interaction of those actors, an agency that retroacts on all different levels of the compositional process.

5. CONCLUSIONS AND OUTLOOK

This article presented an experimental approach in computer music practice that investigates the implications of the agencies involved in composing and performing music with algorithms. I proposed an ecological perspective that understands these agencies as interdependent mechanisms that actively define the artistic process and I described an experimental method through which I artistically investigated these interconnections. I introduced a method of reformulation as a particular form of algorithmic experimentation that I adopted to balance the agencies involved in composing digital synthesis algorithms. This experimentation led to the development of *Else*, a generative system intended for composition, performance and improvisation, whose peculiarity is that its sound outcomes are not implied not intended in its algorithmic formulation.

In future research I would like to pursue the method of reduction, described in Section 3.10, in order to better understand its implications. Even if at present the system has already an extremely condensed form, by pushing this process further I would like to address questions like: what's could be an incompressible algorithmic agent? What is the most basic algorithmic process that exhibits agency? I speculate that continuing this direction a second emergent threshold will crossed, in which the system switches back to an instrumental dimension. Could this coincide with the limit of the minimal complexity needed for agency?

I think that, when focusing on the idea of agency, the perspective on artistic practice that opens up is not one of a *linear* activity that develops after an idea that is conceived *a priori*. It is rather an open process that self-determines itself out of the entanglements and interconnections of the forces involved in its definition. I believe that pursuing these questions, especially with the method of artistic research, could contribute to new critical insights for computer music.

6. REFERENCES

[1] H. Brun. *When Music Resists Meaning*, in *From Musical Ideas to Computers and Back* (1970). Wesleyan University Press, 2004.

[2] L. Döbereiner, *Compositionally Motivated Sound Synthesis*, In *Proceedings of next generation 3.0*, ZKM Karlsruhe, 2009.

[3] L. Döbereiner, *Models of Constructed Sound: Non-Standard Synthesis as an Aesthetic Perspective*, In *Computer Music Journal 35:3*, Fall 2011.

[4] G. M. Koenig. *Genesis of form in technically conditioned environments*. Interface, 16(3), 1987.

[5] Koenig, G. M. 1980, *Composition Processes*, In M. Battier and B. Truax, eds. UNESCO Computer Music: Report on an International Project Including the International Workshop Held at Aarhus, Denmark in 1978. Ottawa: Canadian Commission for UNESCO.

[6] H. H. Rutz, *Marking a Space of Algorithmicity*, in *Proceedings of 4th Conference on Computation, Aesthetics and X (XCoAx)*, Bergamo, 2016.

[7] H. H. Rutz, D. Pirró, *Algorithms That Matter*, https://almat.iem.at, (accessed March 27, 2018).

[8] T. Murail, *The Revolution of Complex Sounds*, in *Contemporary Music Review*, 24[2]:121–135.

[9] L. Parisi, *Contagious Architecture: Computation, Aesthetics, and Space* Cambridge, MA: MIT Press, 2013.

[10] S. R. Holtzman, *A Description of an Automatic Digital Sound Synthesis Instrument*, DAI research report No. 59. Edinburgh: Department of Artificial Intelligence, 1978.

[11] T. Back, D. B. Fogel, Z. Michalewicz, *Handbook of Evolutionary Computation*, Oxford University Press; Lslf edition, April 17, 1997.

[12] G. Briscoe, P. Dini, *Towards Autopoietic Computing*, OPAALS 2010: Digital Ecosystems, 2010.

[13] P. De Loor, K. Manac'H, A. Fronville, *Simulation of abstract autopoietic machine.*, 10th European Conference on Artificial Life (ECAL); ECAL2009 : Darwin Meetsvon Neumann, 2009, Budapest, Hungary.

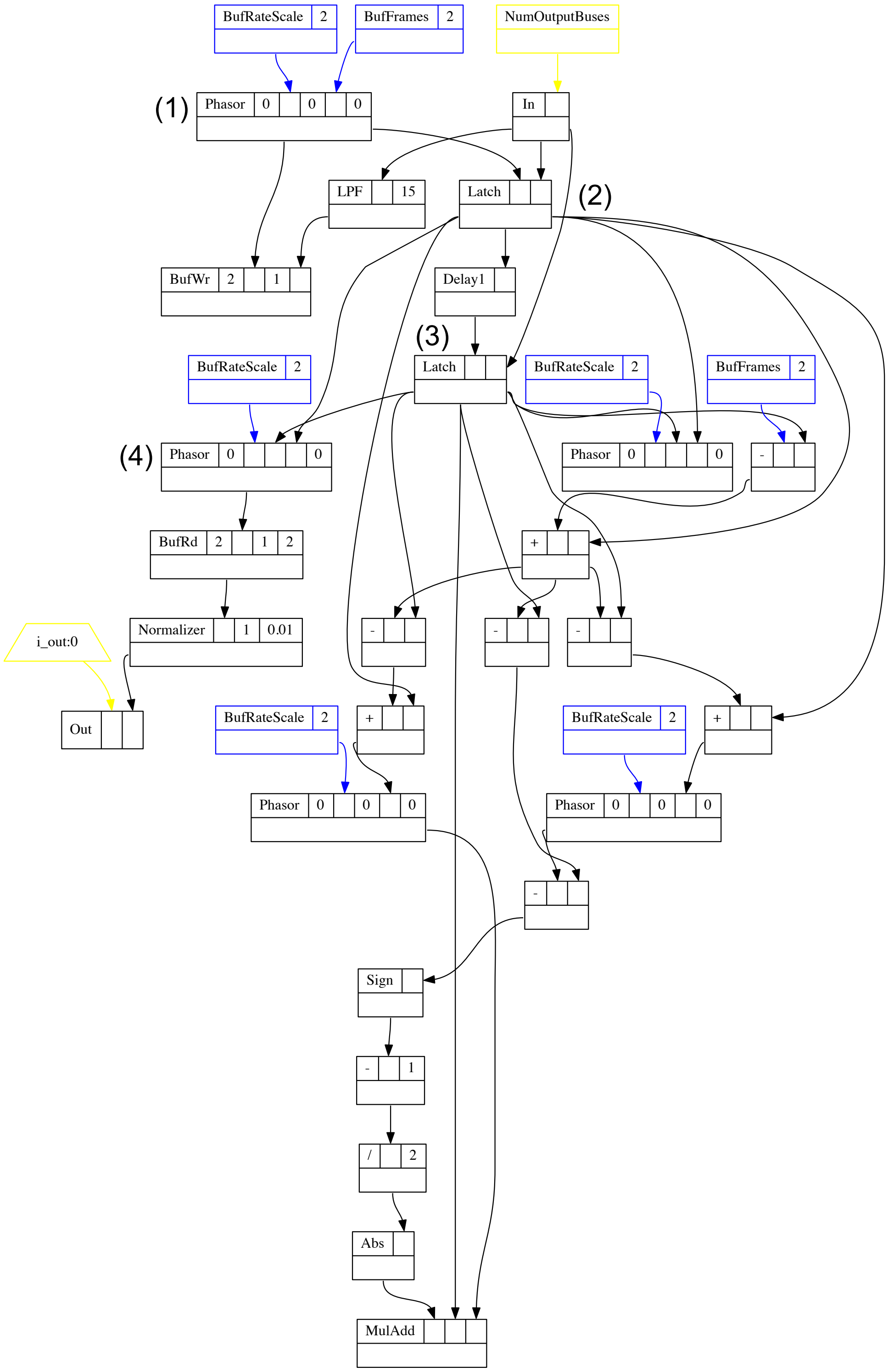

Figure 2. Mechanism after reformulation.