

Degrees of Interpretation in Computer Aided Algorithmic Composition

Jessica Aslan

Univeristy of Edinburgh

j.aslan@sms.ed.ac.uk

ABSTRACT

In 2012 Michael Edwards introduced his open-source composition system, *Slippery Chicken* (*sc*). Since then I have been working with the software, experimenting with the possibilities and limits of its output and identifying its constants and mutations. In this paper I will analyse some of the different compositional methodologies that *sc* offers, tracing its digital fingerprint and examining its persistent presence through degrees of composer and performer interpretation. I will include a discussion of the broad spectrum of opportunities for the parallel generation of ideas and maintenance of each user's compositional voice, not only through choice of input material but flexibility of output formats from the software.

Summarising some current thought on Computer Aided and Algorithmic composition I will attempt to unpick some of *sc*'s design mechanisms, with particular attention to the relationship between form and process of composition when using the software. I will then examine case studies from my compositions with specific reference to *degrees of interpretation*. Firstly, I will present my experiences using the software in a *first degree* approach, which represents unmediated algorithms. Following this I will look at hybrid mediation, *second degree* usage. In this case study *sc* is still directly present through sound file organisation in a fixed-media part, however the notated score is created through aural interpretation of the fixed-media. Finally I will outline the compositional methodology in a *third degree*, fully mediated composition in which I place myself directly in front of the information flow between algorithm and score, meaning no digital (only a perceptual) trace of the software can be found.

1. INTRODUCTION

Slippery Chicken (*sc*) is “a new open-source algorithmic composition system, which enables a top-down approach to musical composition” [1]. Michael Edwards, its creator, describes it as an initially specialised composition software, that has gradually morphed into a more general set of tools. *sc* was initially created to enable Edwards' own compositions, and much of the musical thinking found

in its fabric embodies traits common to his own compositional voice. In his words “it offers a structured method as opposed to a composition software library” [1], however its open source nature means users are free to extract and augment any number of its functions, much like a library. This flexibility means user methodologies can vary greatly, and presents an interesting tool for examining the presence of each software developer's inbuilt musical preferences combined with user intervention.

Since its release, *sc* has been my principal tool for composition. In my time using the software it has been a primary concern that I maintain my own compositional voice, not only experimenting with input varieties (harmonic palettes, rhythmic character, recursive transitions and so on) but stretching the output formats that *sc* lends itself to. Edwards himself aligns his user of *sc* as “firmly in the algorithmic camp” [1] (in terms of Munro's [2] definition). As a user that often mediates algorithm and concrete output I associate my own practice of composition with *sc* alongside Ander's and Miranda's description of computer-aided-composition “where composers manually shape certain aspects of the resulting music” [3]. Therefore, to avoid conflicting terminology I will refer to Christopher Ariza's hybrid expression, Computer Aided Algorithmic Composition (CAAC), which he employs to increase specificity to the often separated definitions of Algorithmic and Computer Aided composition [4]. This will allow me to circumvent any confusion between the two terms, however useful a distinction may be.

1.1 Slippery Chicken in summary

Detailing the idiosyncrasies of composition software is no easy task. Ariza [4] offers some useful descriptors to understand elements of functionality found in CAAC software, and it is useful to offer a short summary of *sc* with these qualifications in mind. *sc* is an open non real-time process model that features an intuitive text (LISP) -based language interface. It offers a wide variety of options for material input and a largely open formatted output, it is ostensibly a “plural idiom affinity...[it] allows the production of multiple musical styles, genres, or forms.” [4], and features full extensibility to the user with some LISP programming skills.

When unravelling the effect of differing input and output material and interior processing, the idea of a plural affinity becomes more complex. Though *sc* is extensible and

Copyright: ©2014 Jessica Aslan et al. This is an open-access article distributed under the terms of the [Creative Commons Attribution 3.0 Unported License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

fairly open, its mechanisms are rooted in Edwards' compositional thinking - particularly when it comes to large-scale form. So though *sc* doesn't restrict the user to a singular approach, some of the inner operations for configuring a complete musical work are sonically quite distinct. Even with an attempt at simple affinity attribution, it is easy to see how definitions identifying traits in CAAC software are hard to secure.

1.2 Process and Form

Unpicking the software contribution to musical form is also tricky, particularly considering the contribution of context to musical perception [4]. In *sc*, user defined input and output are reasonably open, the material itself being the choice of the user, with the shape that it takes (pitch and rhythm sequence palettes, set maps) being determined. Please find more information in the online manual [5]. The character of *sc*, latent within parts of the code more hidden from user view, manifests through processing on input material, the final combination consisting of initial user defined units that are processed within a fixed set of constraints.

The nature of *sc*'s top down approach characterises its output as globally as well as locally organised, with large scale structures created directly through the recombination of pitch and rhythm sequence palettes, with crucial attention paid to transition between sections (see [1] for a detailed description of some transitional features). Because of this *sc* ostensibly avoids Nick Collins' observation of much algorithmic composition software as "stuck in a static moment form, able to abruptly jump between composed sections but unable to demonstrate much real dramatic direction" [6]. In fact, the musical forms that *sc* creates are perhaps one of the most defining properties of the software. A great deal of attention is given to transitioning through subsequent sections often calling on natural processes (L systems, fibonacci transitions) in contribution to the coherence of long term forms.

Practitioners acknowledge varying levels of coherence between form and process - some placing more distance between technical means and artistic output than others. Authors writing on CAAC often use phrases like "piloting the vessel" [7] or employ descriptions of software as "a bicycle, offering mobility to a composer" [3]. These metaphors invoke an analogy of A to B, with the software as an aide to transportation to a final aesthetic object distinct from the means that took it there. For Koenig [8], however, form determines process and process determines form. Thorsen, elaborating on form in more general terms describes it as "The study of how identifiable smaller parts would integrate into greater wholes" [9], this integration, the mechanical processes acting on the smaller parts also making up the form itself.

Nicholas Cook takes care to highlight how intertwined material is with the formal proportions of a work:

Thus, though compositions can certainly create the effect of being well or badly proportioned, this has to do with the qualitative as well as the quantitative aspects of the music; and this is why, when a piece's proportions are faulty, putting it right is likely to involve modifying its content rather than simply cutting out a few measures here or adding an extra beat or two there. [10]

The point he raises is that there are processes that are temporally appropriate to given material. The idea of a piece of music being well proportioned relies not only on abstract schema, but the natural transformation that its smaller elements lend themselves to. He takes this idea further by describing form as "defined by the listener's intentions", meaning that though internal schema may exist they may have little bearing on the perceived form of the final aesthetic object, not unlike Ariza's reference to context as crucial to the perception of form. In other words user material (input and output) has as much influence on the perceived form as the organising processes. Therefore by establishing modes of composer mediation in the process of composition, we can begin to examine formal elements of the work that are strongly influenced by input/output and those that rest more heavily on the software's internal schema.

1.3 Degrees of Interpretation

In order to understand my user influence on the final aesthetic objects, I am classifying my case studies into *degrees of interpretation* (DOIs), indicators of composer mediation related to the output format of *sc*. *First degree interpretation* indicates unmediated output, the algorithm remains untouched post generation for interpretation by a performer. *Second degree* indicates hybrid mediation - I have manipulated some aspect of the output before performance. Finally, *third degree interpretation* indicating complete user mediation of the output format - there is *no digital trace*. These simple distinctions shed light on the flexibility of *sc* as a compositional tool but also bear witness to its influence on structural organisation. A documentation of the user experience will show areas of the software's flexibility but also musical qualities that can potentially persist through any degree of user mediation.

By presenting a user assessment of the software, rather than a developer's explanation I hope to illuminate previously undocumented aspects of the software and shed light on the means of "aesthetic integration" [8] in CAAC. With this in mind I will begin to assess the relationship between my own subjective decisions and those made by the fabric of the algorithm in order to track the musical traces of *sc*. Through varying Degrees of Interpretation, I'm aiming to clarify levels of mediation that existed in the act of creating each case study in order to evaluate *sc*'s contribution to my compositional process.

2. FIRST DEGREE INTERPRETATION

I will examine first degree interpretation with two movements *Labyrinths*, for string quartet and computer, which I created in collaboration with the ISON quartet. Each of the movements draw from separate short stories by Jorge Luis Borges and explore some of the narratorial themes and mathematical paradoxes that he presented. I'll look at the first two movements, *The Garden of Forking Paths (TGOFP)* and *The Circular Ruins* and unfold each compositional process with reference to my mediation of *sc* output.

2.1 The Garden of Forking Paths

I created *TGOFP* through a LISP coded wraparound technique focussing on the multiplication of intervals, with the navigation of the subsequent tonnetz a nod to the literary representation of the infinite found in Borges' story. Here I frame the musical material - creating a function that facilitates the generation of *sc* friendly information.¹

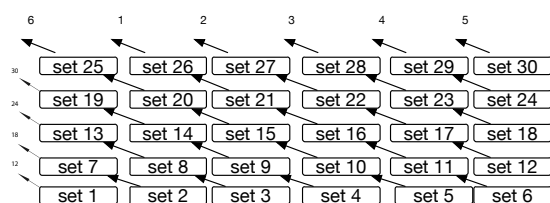


Figure 1. Tonnetz navigation. The harmonic progression infinitely forks to the left.

The harmonic wraparound is the only deviation from a) *sc*'s own code and b) usage as described in the extensive online tutorials and manual, and I did not interpret the output other than in the forms automatically produced so I am ascribing *TGOFP* as a first order *sc* composition. The material as it is played can be generated through a single compilation and I do not mediate the material. The *sc* algorithms specifically generated not only the temporal structure, but carried out the orchestration, and completely assigned all the associated rhythms and harmonies. I have not attempted to bend the output format in any way.

2.2 The Circular Ruins

The Circular Ruins, named after Borges' depiction of the phenomenon of the simulacrum, was formed using a different approach. The idea of the simulacrum and the environmental depiction within the story is important to the fabric of the material - I wanted to evoke an ever shifting instrumental texture through simple material and flexible sound shapes that consistently shift in terms of onset, continent and termination. The realtime electronics become the mirror of the instruments, before eventually engulfing

¹ The harmonic material is pushed through a dense rhythmic palette, using the software's *Rhythm Chains* method. No post generation editing was carried out.

the material completely. I often use spectromorphological analysis as a way to contribute to my understanding of formal coupling in mixed works, and these sound shapes are also a useful method of viewing ensemble material. I used my harmonic wraparound function to generate a new tonnetz (arbitrarily navigated in a circular fashion), and created a very simple rhythmic palette. The emphasis here was the textural change of the ensemble body rather than any particular rhythmic interest (the movement has no time signature).

The interest in this movement is in the timbre and dynamics of the notes, the texture of the ensemble. To harness Smalley's sound shapes I used *sc*'s lilypond graphical notation and added 26 sound shapes as potential articulation. Crucially, I assigned potential parameter changes to these shapes, developing an overall algorithm for the position of each note in the sound shape and their relative dynamics and articulation. For instance a sudden onset might indicate a pizzicato in the first instrument, with additional ensemble notes contributing to the sound body to reinforce each individual shape.

This composition is also first order: a single compilation of my code will create the score that you'll see below for interpretation by an ensemble, but in contrast to *TGOFP* I have incorporated my interpretation of sound shapes and augmented the software to suit my needs. In other words I altered the algorithm but *prior* to the generation of any notes.

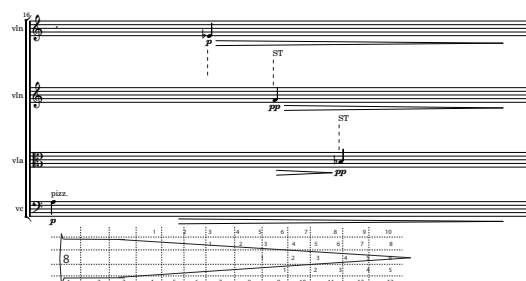


Figure 2. A sound shape and its algorithmically notated representation

Creating both of these works I took advantage of the extensibility of *sc*, an even extracted some of its internal functions to create my own compositional add ons. However, with both these first degree compositions some of the musical qualities found in its functions, particularly the *Rhythm Chains* and L-system transitioning through harmonic progressions are clearly identifiable in the works (musical examples will be presented). I classify both these works as unmediated because though the software may have been altered, the output is accepted without any further editing and the core of the software's mechanism remains intact.

3. SECOND DEGREE INTERPRETATION

Contrasted with my relatively simple approach to generating material in *Labyrinths*, *Mechanica* for violin - Emma Lloyd - and computer, weaves a more complex web. This piece features hybrid mediation, a fixed-media part was produced by *sc* and the instrumental material was formed through composer intervention - a transcription of notes from within the fixed-media part. I am attributing *Mechanica* as second degree intervention.

3.1 Mechanica

Explaining the methodology in this work requires a side step from algorithm to authorship. *Mechanica* began through extraction of recorded samples from seeds of material that I gave Emma, which she then played in an array of unique timbres. This initial step was what gave the piece its clarity and overall character, something that the subsequent algorithm was built to emphasise. This collaborative process complicates the developer/user relationship further - input material is created by a musician and frozen in time through recording. The quality and grain of the work then has relied on the performer, and the resultant aesthetic object is therefore dependent on a third individual. However, as this paper is concerned with *post-generation* mediation, I won't focus further on this aspect.

Once divided and categorised, our samples became the fuel for a fixed-media piece consisting of seven parallel computer parts, consisting of different (though similar) material and made from seven different *Rhythm chains*. The data was exported to *Common Lisp Music* (CLM), an output format fully incorporated into *sc*. Again the foundation of this work is through the software's *Rhythm Chains* algorithm, the rhythmic tendencies perhaps similar to those of *TGOF*, but masked through duplication. This work explores self similarity, the seven slippery chickens all use the same rhythmic information but are called at different speeds, in a canon.

The fixed-media alone is first order - after input of material the piece can be compiled in a single sweep. The instrumental part, however was created through my intervention. From the seven consecutive threads I transcribed a single melodic line - the instrumental part, which Emma plays live alongside the fixed-media. Though the structure of the work and the rhythmic qualities all arise through the algorithm, the instrumental part was borne of my ear, my compositional intervention. The output format is no longer intact and therefore the work is second order, as some algorithmic trace is present, but the piece also relied on output mediation.

4. THIRD DEGREE INTERPRETATION

The final piece that I will examine is *Cantor Dust* for string orchestra. This piece uses *sc*'s L-systems algorithm to digitally augment a traditional Bulgarian folk tune.

4.1 Cantor Dust

I began by recording the tune (see figure 3) on the piano and processing it. Again, much of the grain of the work comes from this initial step of recording and freezing certain acoustic attributes. As the title indicates, self similarity is the central focus, with particular emphasis placed on parameterised DSPs.



Figure 3. Original folk tune

Cantor Dust is another example of *sc* functionality in conjunction with CLM. To create a multi layered fixed-media part from this fragment I processed eighteen different streams of the same recording, each assigned 6 separate DSP parameters: low-pass filter frequency, high-pass filter frequency, transposition, duration, start position in file. These streams began at different frequencies, and progressed through the L-system at different rates. What resulted was a dense cloud of static sound, a fixed piece formed through the layers of evolving musical strands.

Here I interpreted the algorithm through audio transcription. I divided the piece into instruments and notated in detail each prominent frequency and its trajectory through the piece. As the melody was linearly processed, each had a fairly logical direction and as such the fixed-media has a persistent character. This gave me the skeleton of the piece, which I then metamorphosed into a slightly more familiar harmonic form whilst maintaining voice leading and simplified rhythmic relationships.

In performance there is no element of the work implemented through algorithm that I have not actively transformed and reconfigured in some way, therefore it is third order, akin to Essl's notion of an "inspiration machine" [11]. The quality of the software processes most embedded into the final work is the evolving nature of the different musical lines, in particular the pacing and temporal organisation. However, the work is filtered through my ear, my choices made with a very personal background and musical training. What endures is the global architecture, which seems to be highly consistent between each DOI.

5. CONCLUSION

"If one focuses on transitions between moments perhaps the global organisation - the form - also begins to make sense." [6]

Through separating my work patterns into DOIs I have in some way illustrated a level of distinction between the contribution of input (material) and mechanism (process) reflected in different compositional methods. The nature of *sc* means that in some sense the composer is also the

primary listener, and thus able to assess focal points and able to shape the form in a more audibly concrete (rather than abstract) way.

The input and output formats of *Cantor Dust*, *Mechanica* and *TGOFP* are very different. However they are both largely grammatically organised by the *Rhythm Chains* method. The material they both consist of is unrelated, but the rhythmic tendencies on a micro level are arguably parallel, representing some consistency in compositional method - Edwards' own compositional disposition appearing. L-systems and fibonacci sequences are transitioning functions that Edwards has developed extensively, and can be used to structure a work with ease. Regarding larger scale form, the pieces are also comparable in the fact that each exhibits the constant transitional evolution of the material, rather than jumping from moment to moment. This makes sense - the form arises from the context of the material, and the material's suitability in its context. *Rhythm Chains* is a consistent process and inevitably will leave some formal traces of its identity through its process.

The above analyses do indicate a general consistency encouraged by my use of *sc* regardless of input or output, which is its ability to macrostructure a work with logical musical coherence. This is clear when listening to the pieces - each demonstrating evolution of longer musical lines from small input fragments. Though each work entailed differing amounts of mediation, in each the top down structure of *sc* encourages compositional thought towards extended musical lines. The suitability of input material and consequent output format are largely responsible for the final pieces, the impact of innately programmed (in this case transition) functions in *sc* shapes the users' choice of input material - illustrating a continuous feedback loop between software and user. The level of composer mediation of course effects the final aesthetic product, but there are some elements of software that remain musically present even when there is no digital trace of the algorithm.

This paper represents the beginning of what will be a long process of navigation through this rich and powerful musical resource. Assessing future compositions on these terms will help me to understand, develop and share compositional methods, creating a platform for communication regarding composer intervention in CAAC.

6. REFERENCES

- [1] M. Edwards, "Introduction to slippery chicken," vol. Non-Cochlear Sound. Ljubljana: International Computer Music Conference, September 2013.
- [2] G. Munro, "This is art not science," *Leonardo Music Journal*, vol. 7, pp. 77–80, 1997.
- [3] T. Anders and E. R. Miranda, "Interfacing manual and machine composition," *Contemporary Music Review*, vol. 28, no. 2, pp. 133–147, April 2009.
- [4] C. Ariza, "Navigating the landscape of computer-aided algorithmic composition systems: A definition, seven descriptors, and a lexicon of systems and research." San Francisco: International Computer Music Association, 2005.
- [5] M. Edwards. (2014) Slippery chicken manual. [Online]. Available: <http://michael-edwards.org/sc/index.html>
- [6] N. Collins, "Musical form and algorithmic composition," *Contemporary Music Review*, vol. 28, no. 1, pp. 103–114, February 2009.
- [7] C. Ames, "Automated composition in retrospect: 1956–1986," *Leonardo*, vol. 20, no. 2, pp. 169–185, 1987.
- [8] G. M. Koenig, "Composition processes," 1978.
- [9] L. Thoresen, "Form-building transformations: An approach to the aural analysis of emergent musical forms," *The Journal of Music and Meaning*, vol. 4, 2007.
- [10] N. Cook, "Musical form and the listener," *The Journal of Aesthetics and Art Criticism*, vol. 46, no. 1, pp. 23–29, 1987.
- [11] K. Essl, *The Cambridge Companion to Electronic Music*. Cambridge University Press, 2007, ch. Algorithmic Composition.