

Experimental Humanities

An Adventure with Lovelace and Babbage

David De Roure
Oxford e-Research Centre
University of Oxford
Oxford, United Kingdom
david.deroure@oerc.ox.ac.uk

Pip Willcox
Centre for Digital Scholarship
University of Oxford
Oxford, United Kingdom
pip.willcox@bodleian.ox.ac.uk

Abstract—The development and innovative application of digital research methods in humanities disciplines, characterised as Digital Humanities or e-Humanities, is an established feature of the e-Science and e-Research landscape. Typically these digital methods enable existing research questions to be tackled in new ways, at a scale and speed that transcend manual methods. In this paper we present a different approach to the application of digital techniques to humanities research, a branch of *experimental humanities* in which digital experiments bring insight and engagement with historical scenarios and in turn influence our understanding and our thinking today. We illustrate this through a series of experiments and demonstrations inspired by the work of Ada Lovelace and Charles Babbage, including simulation of the Analytical Engine, use of a web-based music application, construction of hardware, and reproduction of earlier mathematical results using contemporary computational methods.

Keywords—Ada Lovelace; Charles Babbage; Leonard Euler; Analytical Engine; algorithmic composition; Digital Humanities; Reproducible Research

I. INTRODUCTION

We have conducted a series of experiments to explore Ada Lovelace’s observation that Babbage’s Analytical Engine “...might compose elaborate and scientific pieces of music of any degree of complexity or extent” [1]. These experiments involve simulation of the Analytical Engine, design and use of a web-based music application, design of hardware, and reproduction of earlier mathematical results using a computer algebra system. While both ‘digital’ and ‘humanities’, our work is distinct from much work in the Digital Humanities arena, and owes something also to creative computing.

This work is supported by: *Fusing Semantic and Audio Technologies for Intelligent Music Production and Consumption* funded by the UK Engineering and Physical Sciences Research Council (EPSRC) under grant number EP/L019981/1, a collaboration between Queen Mary University of London, University of Nottingham and University of Oxford; and *Transforming Musicology*, funded by the UK Arts and Humanities Research Council under grant AH/L006820/1 in the Digital Transformations programme, in collaboration with Goldsmiths University of London.

While some aspects of this work have contributed to a range of publications, including e-Science [2], Semantic Web [3] and Mobile HCI [4], this is the first comprehensive account of our experiments and demonstrations to date and invites reflection on the methodology. Our research is clearly practice-led, as described by Archer [5]: “There are circumstances where the best or only way to shed light on a proposition, a principle, a material, a process or a function is to attempt to construct something, or to enact something, calculated to explore, embody or test it.” Although this may not be the way in which e-Science work is typically presented, we propose this as a valid methodology for our endeavour, and an approach that might be discussed in e-Science more broadly.

The next section sets the scene in the 19th century, providing the historical context and motivation of our work, and is followed in Section III by an account of the series of experiments and demonstrations that we have conducted. In Section IV we reflect on our approach, particularly in the context of *Experimental Humanities*.

II. LOVELACE AND BABPAGE

We focus on a period in the early 1800s, the time of the punch-card operated Jacquard loom and the first electrical telegraph. For a comprehensive account of Lovelace’s life we refer the reader to the many biographies, such as [6].

Augusta Ada Byron (from whose later married name, Ada King-Noel, Countess of Lovelace, we take her more usual style, Ada Lovelace) was born in December 1815 to Anne Isabella Milbanke, 11th Baroness Wentworth, and George Gordon Byron, 6th Baron Byron. After a year of turbulent marriage, Milbanke left the family home for her parents’, taking with her the month-old Lovelace; neither saw Byron again.

Milbanke was intelligent and received a broad, advanced education including tuition by William Frend, a former fellow of Christ’s College, University of Cambridge. It seems likely that the scientific approach to education she favoured for her daughter was influenced both by her personal passion for mathematics, and in reaction against her estranged husband, whose Romanticism championed individual response over rationality and science, and whose perceived mental ill-health she hoped to avoid cultivating in Lovelace.

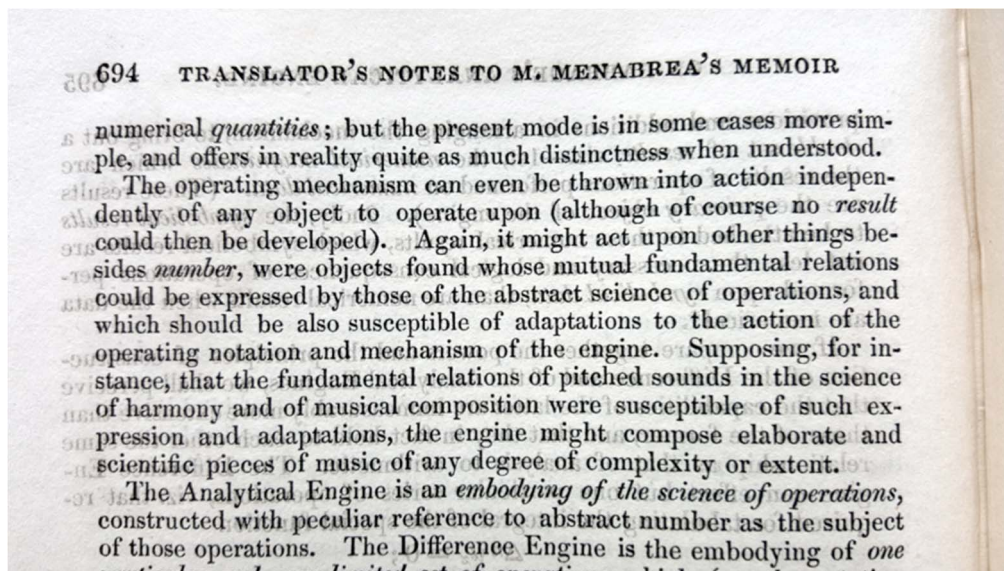


Fig. 1. Image of part of Lovelace's note A, in her translation of Menabrea's transcription of Babbage's 1840 Turin lecture (photo: Pip Willcox, from Taylor's Scientific Memoirs [1] provided by Magdalen College Library, University of Oxford).

Lovelace's enquiring mind and particular affinity with mathematics thrived in her somewhat remote upbringing. Her achievements extended into many areas of study, including music: she played instruments including violin and piano, and her favourite, the harp. She was tutored at home as a child and pursued lifelong education, including through conversation and correspondence with notable mathematicians amongst whom were Mary Somerville, William Frend (her mother's former tutor), and Augustus de Morgan. Later acquaintances included Michael Faraday, Charles Dickens and Charles Wheatstone.

Lovelace's imaginative engagement with science, and especially mathematics, set her in good stead to understand the workings and potential of Charles Babbage's Difference Engine. In 1833, and during her first 'season' in society as a 17-year-old, she formed part of a group to which Babbage demonstrated a prototype of the difference engine, and her response to it appears to have sparked the friendship between them that lasted until her death in 1852, aged 36. Lovelace achieved a deep understanding of the design for Babbage's Analytical Engine, his proposed but never constructed steam-powered general purpose computer.

For such a prolific author, Babbage did not publish much about his Analytical Engine. He gave one lecture on it, at the University of Turin in 1840. A version was published in French by an audience member, Luigi Federico Menabrea (engineer and mathematician, who later served as a lieutenant-general during the unification of Italy, and subsequently as Prime Minister of the unified country). In 1842, Lovelace was commissioned by Wheatstone to translate Menabrea's paper into English and she added extensive notes, more than trebling the length of the publication. Famously these notes contain the first algorithm designed for execution by a general purpose computing machine, for calculating Bernoulli numbers, and this is often described as the first computer program.

They also contain two important ideas which have been taken up in subsequent academic debate and widely cited, and which inform our work:

1. "Supposing, for instance, that the fundamental relations of pitched sounds in the science of harmony and of musical composition were susceptible of such expression and adaptations, the engine might compose elaborate and scientific pieces of music of any degree of complexity or extent." (note A in [1], see Figure 1). This is often cited within the computers and music domain.
2. "The Analytical Engine has no pretensions to *originate* anything. It can do *whatever we know how to order it to perform*. It can follow analysis; but it has no power of anticipating any analytical relations or truths. Its province is to assist us to making available what we are already acquainted with." (note G in [1], Lovelace's emphasis). This is cited in debates around computers and creativity, in particular by Turing in the "imitation game" paper [7] and, in turn, Margaret Boden in her "Lovelace questions" [8].

December 2015 saw the 200th anniversary of Lovelace's birth, and a major symposium was held to mark the occasion. As part of the anniversary celebrations, composer Emily Howard conducted a performance by musicians from the Royal Northern College of Music of one part of her Lovelace trilogy, 'Ada sketches' [9]. By background Howard is a mathematician and computer scientist, and uses mathematics in her compositional process. The sketch features a mezzo soprano singing a libretto by musicologist Laura Tunbridge, and enacts the moment the character of young Lovelace grasps the potential of the Difference Engine to calculate powers. The performance was accompanied by an explanation of the music by Howard and of the mathematics by Lasse Rempe-Gillen (University of Liverpool). The audience was invited to turn 'Numbers into Notes' using simple algorithms.

In discussing this public event at the symposium we also developed a thought experiment: had Lovelace lived longer, and had Babbage successfully built the Analytical Engine, what might have happened in pursuit of Lovelace’s musical observation? [10] This was the motivation for our sequence of experiments and demonstrations, which we describe in the following section.

These experiments involved understanding the mathematical and musical context of the time. While the mathematics of the early 19th century remains familiar today, there are results we use now that were unavailable then. For example, research in computers and music makes extensive use of Markov chains, and Markov was not born until 1856. However, aleatoric techniques, like the musical dice game, had been established before Lovelace’s time. Today we cite Helmholtz’s work on the consonance of musical tones, but his *Sensations of Tone* [11] was not published until 1875; Lovelace might instead have had access to the mathematical definitions of consonance (or ‘degree of harmoniousness’) in Leonard Euler’s 1739 work *Tentamen* [12].

Lovelace’s wider musical context, we might assume, included familiarity with the classical repertoire, church music, and also social dance music, where the musical form is constrained by choreography. This was the time of Romanticism, a movement that emphasised creativity and the individual, often seen as a response to the Scientific Revolution and Age of Enlightenment it succeeded. The Lovelace quotations, however we might interpret them today, must be understood to have arisen in this context.

III. EXPERIMENTS AND DEMONSTRATIONS

A. Using the Analytical Engine simulator

Babbage’s Analytical Engine remains unconstructed to date, but we can simulate the execution of programs on it digitally, based on the detailed accounts of the engine’s design provided by Babbage, Menabrea, and Lovelace.

In this first experiment, which was presented at the Ada Lovelace Symposium, we set out to generate a number sequence that could then be performed as music, perhaps mechanically or by human players. Our hypothesized workflow was that the Analytical Engine would run a parameterized program to generate a number sequence, and parts of this sequence would then be given to different instruments. Inspired by the use of punched cards in the Jacquard loom and in the proposed Analytical Engine, we envisaged ‘piano rolls’ generated by the engine. The numbers are strictly faithful to nineteenth-century mathematics, while human intervention decides the algorithmic parameters and the mapping of the numbers to notes and instruments.

We made a decision at this point to use a Fibonacci sequence for demonstration purposes as it is well known, and we took advantage of the mathematical property that reducing a Fibonacci sequence using modulo arithmetic results in a periodic sequence, as noted by Lagrange in 1774. This is called the Pisano period (see <https://oeis.org/A001175>). For example, when Fibonacci sequences are reduced modulo 5 through to 128 (being the range of numbers of notes we might work with), the Pisano period varies between 10 and 500, with a higher

number of occurrences of shorter periods. The periodic nature was thus apparent in repetition of the music during playback (e.g. at 1–4 notes to the beat, 60–120 beats per minute). Figure 2 shows a histogram of the Pisano periods, generated using the Wolfram Mathematica symbolic computation software which we used as a tool to assist design throughout our experiments.

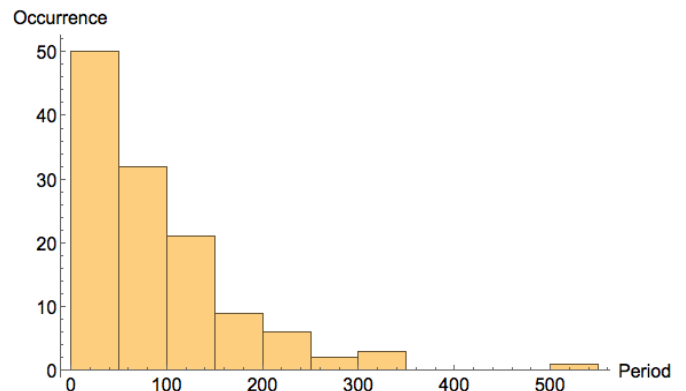


Fig. 2. Histogram showing the distribution of periods when the Fibonacci sequence is reduced by modulus values between 5 and 128, i.e. Pisano Periods.

For the demonstration we used the Fourmilab simulator (<https://www.fourmilab.ch/babbage/contents.html>) by John Walker, which is a Java program that simulates the Analytical Engine. Since the Analytical Engine was never built, it is not possible to provide an authentic emulation, but rather a simulation based on the available designs, John Walker writes.

While the 1837 description of the Engine had 40 decimal digit capacity in the “Store”, and the “Mill” could accommodate the 80 digit product of two numbers, Babbage’s later account in his memoirs describes 50 and 100 digit capacities respectively [13]. This clearly exceeds what we might represent in 64 bit binary, for example: $10^{50} - 1$ requires 167 binary digits, and the Mill would need 333 bits.

Learning to use the simulator requires understanding the nomenclature of the Analytical Engine (such as the notions of analyst, attendant, cards, mill, store, and printer) but otherwise is not dissimilar to the assembly language programming that was later taught in the mid twentieth century. Figure 3 shows a fragment of an execution trace, using a more recent version of the simulator which runs in a browser (using HTML5 and Javascript), illustrating the architecture and operations.

Attendant's Log

```
Card: 27. (Analyst:27) -
Card: 28. (Analyst:28) L4
Store: Mill <= V4(2)
Card: 29. (Analyst:29) L1
Store: Mill <= V1(82)
Mill: 2 - 82 = -80 Run up
Card: 0. (Attendant:1) C8723
Card: 8. (Analyst:8) /
Card: 9. (Analyst:9) L2
Store: Mill <= V2(2)
Card: 10. (Analyst:10) L0
Store: Mill <= V0(35)
```

Fig. 3. Part of the trace of a program generating the first 82 numbers of the Fibonacci sequence mod 35 starting at 1,1. This calculation involves 246 Mill operations and 703 Store operations.

We generated sequences with all numbers mapped to notes (e.g. using chromatic or diatonic scales), and then with a variety of audiences we asked people to identify musical themes that they picked out. We then defined the number and time ranges to isolate these themes and mapped them to different instruments using the Logic Pro digital audio workstation (DAW) software, which enabled us to use high quality samples of instruments similar to those used in Lovelace's time. We have produced several pieces of music in this way, and our demonstration theme, which is based on Fibonacci mod 35, has also been used as a basis for a jazz number. In August 2017 Alan Chamberlain (University of Nottingham) performed a live composition entitled "The Gift of the Algorithm: Beyond Autonomy and Control" based on fragments produced by several contributing musicians, at the *Audio Mostly 2017* conference in London, UK.

B. Numbers into Notes on the Web

The success of the simulator demonstration encouraged us to produce an interactive tool which would enable people to conduct their own explorations, using a variety of algorithms, to stimulate discussion of mathematical calculations on the Analytical Engine, contemplating what Lovelace and Babbage might have done at the time if the engine had been built. The single page web app was coded in JavaScript using HTML5 Web Audio, and is available on <http://numbersintonotes.net>

The example algorithms are summarised below, illustrating their reason for inclusion:

- **Fibonacci**, where the first two numbers can be specified (e.g. if we start with 2, 1 we get the Lucas numbers 2, 1, 3, 4, 7, 11, 18, ...). The generalised sequence is described mathematically by the relation $F_{k,n}+1 = kF_{k,n} + F_{k,n-1}$
- **Powers** (exponentiation), the sequence of powers of 2 or any other specified base. This is an interesting example because the exponentiation calculation on the Analytical Engine can use the squaring method to reduce computational complexity; e.g. $2^{36} = (2^2)^{18} = \dots = ((2 \cdot ((2^2)^2)^2)^2)$. It is also the opening example in the Tunbridge's libretto of Emily Howard's 'Ada sketches'.
- **Pi** was calculated by hand to 100 decimal places in 1706 using Machin's algorithm (and hence we note that algorithms existed before computers). The same algorithm is used here. Pi is believed to be a 'normal number', which means any possible musical sequence appears somewhere in its digits. When Charles Babbage's son Henry constructed part of the Mill of the Analytical Engine in 1910, it was used to calculate multiples of pi to 28 digits.
- **Golden Ratio**, the number that the ratio of successive Fibonacci numbers converges to (1.618...), which can be calculated on the Analytical Engine while generating the Fibonacci sequence. It can also be calculated using a square root function to determine $(1 + \sqrt{5})/2$. The square root algorithm we use today can be coded on the Analytical Engine, and although known as Newton's method it goes back to 1st century Greece or earlier Babylonian times.
- **Prime numbers**, calculated using the Sieve of Eratosthenes, but raising the question as to how this might have been coded for the Analytical Engine. The size of the

sieve is estimated using a 1798 result by Legendre, and in those days 1 was considered to be a prime.

- **Pseudorandom numbers** can be generated on the Analytical Engine, but the algorithms have been developed in the twentieth century, so how might Lovelace have done this? Also, what forms of external entropy might have been possible? Musical dice games (*Musikalisches Würfelspielen*) were popular in the 1700s. Today we analyse and generate music using probabilities of note and chord transitions, using a mathematical approach due to Markov (born in 1856).
- **Factorials** usefully demonstrate the number of digits handled by the Analytical Engine. $20!$ can be stored in a 64 bit integer before we run out of bits, but the analytical engine could go up to $41!$ with its support for 50 decimal digits. The notation for factorial, e.g. $20!$, was introduced in 1808.
- **Bernoulli numbers**, the algorithm for which Lovelace is most famous. Here we generate the numerator and denominator sequences separately. Only short sequences are generated because it is quite computationally intensive, using a rational arithmetic library, which raises the question: how long would the algorithm have taken to run on the Analytical Engine?

Figure 4 shows the piano roll display for Fibonacci mod 35, with a fragment selected for export. The final stage of this workflow is to export the musical fragment in various formats. The tool generates music notation (using the Lilypond music engraving software), as well as standard MIDI (Musical Instrument Digital Interface) files.

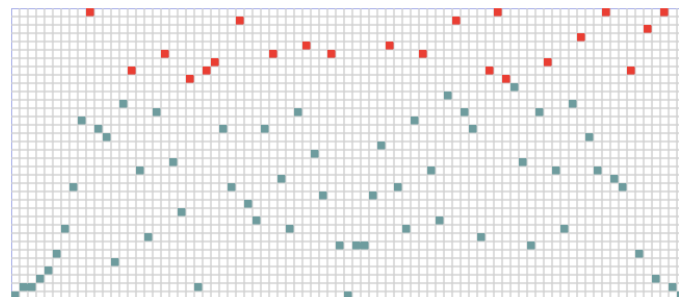


Fig. 4. Piano roll display from the Numbers into Notes application. The vertical axis represents the increasing note number from bottom to top, and the horizontal axis represents increasing time from left to right. The pattern is Fibonacci modulo 35 (period 80) with an upper theme selected.

Other outputs are designed to enable someone to understand how the fragment was generated, or indeed to regenerate it using different tooling; i.e. to reproduce the results of the experiment. One of these is W3C PROV-N (<https://www.w3.org/TR/prov-n/>), from which a SVG (Scalable Vector Graphics) visualization is generated (see Figure 5) using the translator function in the Southampton Provenance Suite (<https://provenance.ecs.soton.ac.uk/>). We also provide an automatically generated natural language description of the algorithm parameters, mapping, and selection.

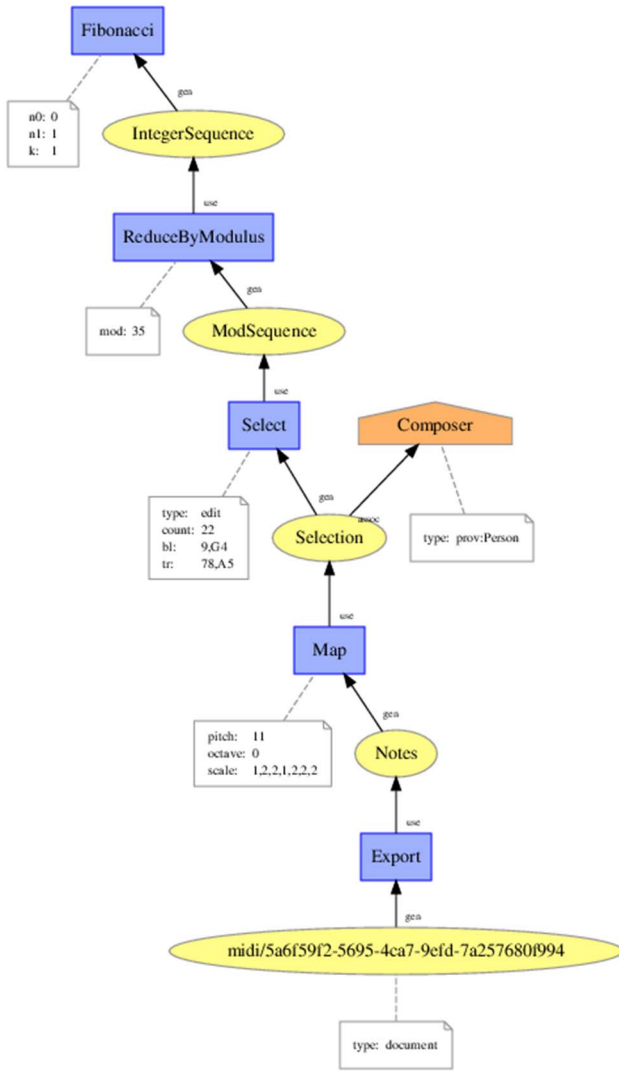


Fig. 5. The provenance graph generated by visualising the PROV-N export from the Numbers into Notes Web application.

C. The Arduionos

The next part of our thought experiment was to ask “what would Lovelace do today?” The proposed analytical engine architecture was impressive even by today’s standards. 50 decimal digits of precision aside, today the algorithms run readily (and, for music purposes, in real time) on the processors of open source Arduino hardware, such as an 8 bit microcontroller, or 32 bit ARM.

Our first experiment was to build the “Arduino analytical engine”. With a slave audio processor (VS1053) and 3W amplifier to drive speakers, we replicated the Numbers into Notes tool as a small standalone “music engine”, depicted in Figure 6.

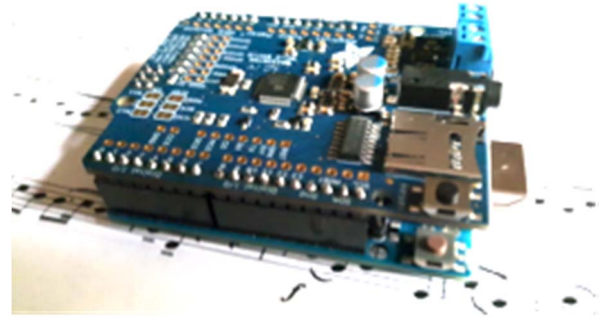


Fig. 6. One of arduinos constructed for the “Fibonacci orchestra”.

Had Babbage been successful in funding he might perhaps have built one Analytical Engine, but imagine more than one, networked. This is not inconceivable. Charles Wheatstone, an acquaintance of Lovelace and Babbage, patented the telegraph system in 1837, and a successful system was installed on the Great Western Railway over a 21km distance (from Paddington station to West Drayton) in 1838. Could the networking techniques of the telegraph have been used in conjunction with the Analytical Engine?

In a more recent experiment with arduinos we have run the algorithms on the “Teensy” device. The model we used, a Teensy-LC, provides an ARM Cortex-M0+ processor running at 48 MHz. Significantly this device natively supports MIDI over USB, which means the devices, which are as small as a USB memory stick, can be plugged directly into any USB interface that is expecting a MIDI keyboard. This enabled us to distribute our algorithms to colleagues in physical form, for use in a variety of demonstrations. An unexpected outcome of this exercise was the social “gifting” behaviour, whereby people can give algorithms to others as physical gifts, and they can be passed on.

Hence we have considered what might be possible with multiple engines that have a means of communication. For our first experiment we built standalone arduinos which could then be located around a physical space and sound simultaneously, in what we nicknamed the “Fibonacci Orchestra”. Here we exploit a musical outcome of our earlier work mapping the output of the Analytical Engine simulator: multiple sequences played alongside each other generate harmonies, and this effect is richer when sequences are played at different but geometrically related tempos (e.g. 120, 60, 30 beats per minute).

The resulting harmony sequences are experienced according to the location and orientation of the listener relative to the devices. In a sense, each human configures the experience in the space for subsequent visitors, hence the performance is co-created. We presented this work [4] but we were limited to small numbers of devices, and interested to understand what might happen with greater numbers. For this we created a simulation in the NetLogo agent based simulator (<https://ccl.northwestern.edu/netlogo/>), illustrated in Figure 7. The simulator enables us to explore emergent behaviours under different algorithms and parameterisations.

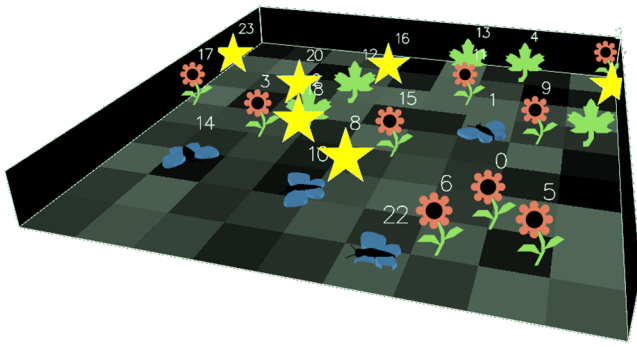


Fig. 7. A simulation of multiple Fibonacci engines, using NetLogo. In this example, spatial location is used so that each engine can sense the audio output of neighbours within a given radius.

D. Mechanical augmentation

The inspiration for this experiment was a visit to the Mechanical Music Museum at North Leach in the Cotswolds, UK, where we learned of the mechanical music box industry that was becoming established at the time of Lovelace. Subsequently the consumer became more interested in devices for the reproduction of recorded music, for example through the phonograph invented in 1877.

Through discussions with colleagues in the FAST (Fusing Audio and Semantic Technologies) project, we considered how the mechanical music boxes might have developed further. We also considered what might happen if the mechanics of the music box came together with the instrument being played, in a mechanically and mathematically augmented instrument.

To explore this we have made our algorithms available through a well established user interface; i.e. a piano keyboard. The ambition was to facilitate creative use of the mathematical algorithms during performance, by creating a new kind of instrument that is in part automated mathematically. We achieved this by coding the algorithms in JavaScript which we run in the Logic Pro DAW, using the “Scripter” plugin. The Scripter processes incoming MIDI and/or timing data, and this processing occurs ahead of any audio generation. Hence the scripts can respond to incoming note events and transform them.

Our first Scripter experiment, called “Fibber”, used the Fibonacci sequence, so that pressing a key triggered a sequence according to predefined parameters (see Figure 8). The sequence decays and the number of notes depends on the velocity at which the key is pressed. In particular this interface lets the performer play one sequence against another and experiment with multiple simultaneous sequences, starting on different notes and with various timings, also using different instrument samples. In use this feels like playing a sampler, but each sample is being generated algorithmically in real time. By mechanical analogy, it is as if pressing each key spins a disk which generates a sequence.

Our second Scripter experiment was the “Algorithmic Arpeggiator”. This uses algorithms to generate notes selected from the set of keys depressed simultaneously, where the order is chosen according to the algorithm (see Figure 9). The particular algorithm of interest in this experiment was a

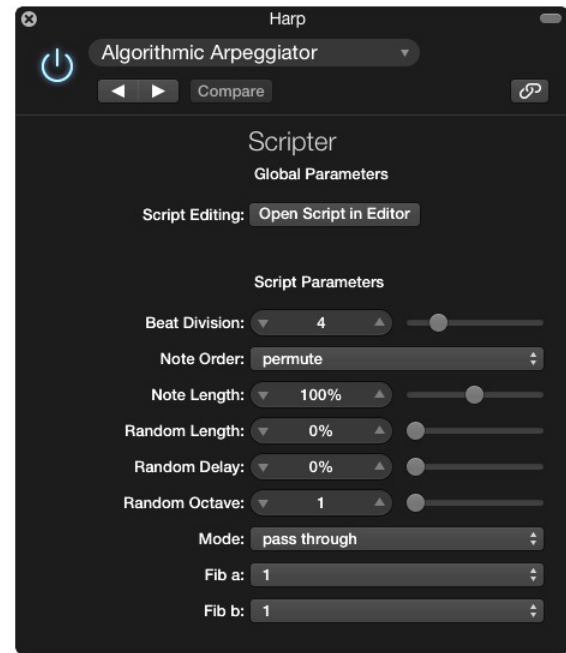


Fig. 8. The user interface for the Fibber experiments, with parameters for a generalized Fibonacci sequence reduced by modulus arithmetic.

permutation algorithm designed to replicate bell-ringing practice, drawing on the important 1668 work *Tintinnalogia, or, the Art of Ringing Wherein is laid down plain and easie Rules for Ringing all sorts of Plain Changes* [14]. This book provides a comprehensive introduction, which is scientific in style. From the introduction to *Tintinnalogia*:

“...for within these Fifty or Sixty years last past, Changes were not known, or thought possible to be Rang: Then were invented the Sixes, being the very ground of a Six score: Then the Twenty, and Twenty-four, with several other Changes. But Cambridge Forty-eight, for many years, was the greatest Peal that was Rang or invented; but now, neither Forty-eight, nor a Hundred, nor Seven-hundred and twenty, nor any Number can confine us; for we can Ring Changes, Ad infinitum.”

In contrast to MIDI arpeggiators, which take a chord and produce a rapid series of notes based on that chord over multiple octaves, the algorithmic arpeggiator has a pass through mode whereby keys pressed are sounded immediately and continue to sound by algorithmic selection as long as the keys are held down. This gives the performer a degree of creative control, as if they and the algorithm were performing together. Again, it is as if a mechanical selector is running behind the scenes.

An extension of this work would be to use a *Disklavier*—an acoustic piano fitted with sensors in order to record the precise movements of keys, and also with electromechanical solenoids to press keys automatically as in a self-playing piano (pianola). Benford *et al* have developed a related system for recognising and responding to musical trigger phrases [15].

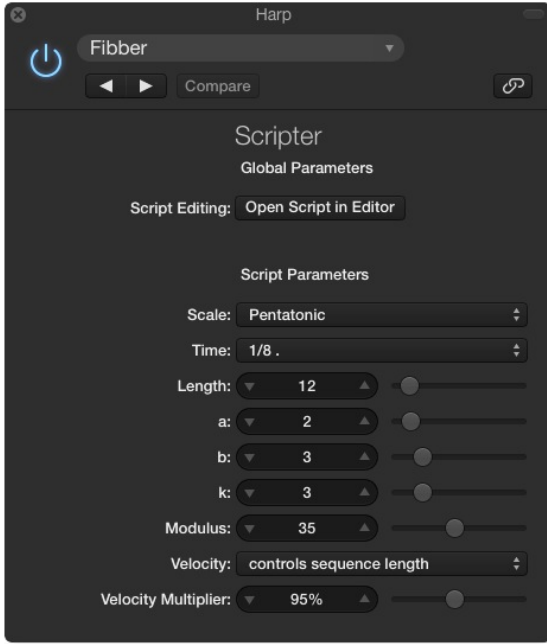


Fig. 9. The user interface for the Fibber and Algorithmic Arpeggiator experiments

E. Reproducing Euler

Our final experiment is an exercise in reproducibility that arose during our exploration of mathematics and music at the time of Lovelace. Today we often refer to the consonance curve from Helmholtz's 1895 *On the sensations of tone* [11]. Before Lovelace, the *suavitatis gradus* (degree of harmoniousness) function for calculating consonance was defined by Leonhard Euler in his 1739 *Tentamen novae theoriae musicae ex certissimis harmoniae principiis dilucide expositae* [12].

In order to code Euler's consonance calculation, we aimed to reproduce the mathematics in *Tentamen*. In fact Euler does not present an equation for the Gradus function, but describes how the function is obtained and applied. Helmholtz also provides a definition in a footnote in [11]. Hence we defined the equation based on the examples in the text, and tested this by reproducing tables in the book.

Expressed mathematically the function is:

$$\Gamma(n) = 1 + \sum k_j (p_j - 1) \text{ where } n = \prod p_j^{k_j} \quad (1)$$

In words, any integer can be represented as the product of prime factors. To determine the Gradus we take the sum of the prime factors, subtract the number of factors, and add 1. Note this includes repetition (multiplicity) of factors. The number of prime factors is also known as $\Omega(n)$, which is the maximal number of terms in any factorization of n .

We published the integer sequence in the *On-Line Encyclopedia of Integer Sequences* (OEIS), which involved peer review of the entry, with other mathematicians reproducing the calculation. Here is part of the definition that appears in OEIS as sequence A275314 (see <https://oeis.org/A275314>):

FORMULA

If $n = \text{Product}(p_j^{k_j})$ then $a(n) = 1 + \text{Sum}(k_j * (p_j - 1))$.
 $a(n) = A001414(n) - A001222(n) + 1$.

EXAMPLE

For $n = 5$ the prime factors are 5 so $a(5) = 1 + 4 = 5$.

For $n = 6$ the prime factors are 2, 3 so $a(6) = 1 + (1 + 2) = 4$.

MATHEMATICA

$\text{Gradus}[n_] := \text{Plus} @@ (\text{Flatten}[\text{Table}[\#1, \{\#2\}] \& @@@ \text{FactorInteger}[n]] - 1) + 1$

We successfully reproduced the Gradus table on page 41 of *Tentamen*, and we have also reproduced the table on page 61 which identifies whole number ratios with gradus from 2 to 10. Using Mathematica, we found that the gradus 10 ratio 2:21 is missing from the published table. Whether this is an omission by Euler in the calculation, or an error in bringing it to the printed page, the extent and accuracy of manual work by Euler remains impressive.

The computational complexity of integer factorization would have limited what was achievable on the Analytical Engine. Even now the 50 digits of the engine would be challenging: for example, we have used HPC capability when conducting experiments in Mathematica with factorisation for integers beyond 10^7 .

One of the purposes of Babbage's Difference Engine was to produce tables, e.g. logarithms and trigonometric functions, and indeed his tables of logarithms were used by the British Admiralty. Similarly the Analytical Engine would have produced tables of extraordinary accuracy, though subject to human transcription. We have worked through the process of a musical scenario using such a capability: if the Analytical Engine were to produce tables of the ratios corresponding to the notes of an equally tempered scale (e.g. for a 12 note scale, these are the decimal values of $2^{1/12}, 2^{2/12}, 2^{3/12}, 2^{4/12}, 2^{5/12} \dots 2^{11/12}$), we could then identify mechanically, or indeed by human inspection, which numbers of divisions of the scale give the best approximation to the simple whole number ratios of consonant intervals. For example, if we identify the sequence of improving approximations to the ratios 3/2 and 4/3, we obtain this sequence of equal divisions of the octave:

2, 3, 5, 7, 12, 29, 41, 53, 200, 253, 306, 359, 665, ...

The same sequence is obtained from the denominators of increasingly better rational approximations to $\log 3 / \log 2$ (see OEIS A060528). There is a pattern to this sequence, whereby each term equals the current term plus one or more of the previous terms, but it is not immediately apparent whether this self-accumulating property would hold as the sequence continues.

One of the capabilities of the Analytical Engine would have been to test such a property beyond what could have been achieved by manual calculation. Computer assisted proof was popularized over a century later, with the verification of the four colour theorem in 1976.

IV. REFLECTIONS

Using digital approaches we have prototyped software and hardware to investigate Lovelace's theory of creative computing. The web-based tool we developed, Numbers into Notes, has been used to engage the public with these ideas, prompting the audience at musical performances of Howard's 'Ada sketches' to create their own digital compositions. Through making, through prototyping and co-design, we engage first-hand with the thought processes Lovelace and Babbage recorded. We continue to explore: for example, what might have resulted from conversations between Lovelace and Darwin?

We have engaged with audiences in a series of events which have brought an important element of co-design to our work. These include:

- Ada Lovelace Symposium (Dec 2015);
- Digital Music Research Network (DMRN), Queen Mary University of London (Dec 2015 and 2016);
- 'Ada sketches', Royal Northern College of Music (Jul 2016);
- ACM MobileHCI conference, Florence (Sep 2016);
- Moore Institute series of Digital Scholarship Seminars, National University of Ireland Galway (Nov 2016);
- Digital Humanities Association of Southern Africa conference, Stellenbosch University (Jan 2017);
- Oxford International Women's Festival (Mar 2017);
- Centre for Creative Computing, Bath Spa University (Mar 2017);
- Digital Humanities at Oxford Summer School (Jul 2017);
- Digital Humanities 2017, Montreal (Aug 2017);
- Interaction, Instruments and Performance workshop at Audio Mostly, Queen Mary University of London (Aug 2017).

Is this Digital Humanities? Clearly our work is both *digital* and treats subjects within the *humanities* space, and falls in the scope of digital humanities in its broadest conception. However, much of the activity that is described as digital humanities today majors on working with digital content (digitised and born-digital) using computing tools to support traditional humanities pursuits and to deliver digital outputs. With our approach we can draw an analogy with, amongst other fields, *experimental archaeology* which generates and tests archaeological hypotheses by replicating or approximating the feasibility of performing various tasks using methods and technologies available in their contemporary contexts. Hence we can frame our work as a branch of *experimental humanities*.

We noted in the introduction that our approach is practiced. It also owes something to speculative design and the use of *design fictions*, in that it creates scenarios narrated through designed artifacts as a way to facilitate and foster critical debate. Whereas this methodology is typically used to explore possible futures, we are applying it in a historical context.

The process of designing and prototyping the series of experiments and demonstrations we outline here resides in what Galey and Ruecker describe as the "messy middle ground between interpretation and making". [14] Here we explicate our critical interpretation of Lovelace's description of the potential processes of the Analytical Engine, using digital tools to explore her thought processes. Our work exploring Lovelace's description of the Analytical Engine's potential for aiding human creativity is a critical interpretation of her

writing. Having demonstrated her ideas through simulation, reifying our argument through the digital prototypes described here, we have extended them to include the Engine as partner in co-creativity.

ACKNOWLEDGMENT

We are grateful to many colleagues for their inspiration, advice and encouragement, including Alan Chamberlain, Tim Crawford, Emily Howard, Andrew Hugill, Graham Klyne, Ursula Martin, Sydney Padua, Michelle Phillips, Lasse Rempe-Gillen, and Marc Webber. We are also grateful to our audiences for engaging in the co-creation process.

REFERENCES

- [1] A.A. Lovelace (Trans.) Sketch of the analytical engine invented by Charles Babbage, with notes by the translator. In Scientific Memoirs, Selected from the Transactions of Foreign Academies of Science and Learned Societies, Vol. 3, 1843, pp. 666-731, volume 3. Richard and John E. Taylor, Red Lion Street, Fleet Street, London. Translation of 'Notions sur la machine analytique de M. Charles Babbage' by Luigi Federico Menabrea, in Bibliothèque universelle de Genève. Nouvelle série 41, pp. 352-76.
- [2] D. De Roure, G. Klyne, K. R. Page, J. Pybus, D. M. Weigl, M. Wilcoxon and P. Willcox. "Plans and performances: Parallels in the production of science and music," 2016 IEEE 12th International Conference on e-Science (e-Science), Baltimore, MD, 2016, pp. 185-192. doi: 10.1109/eScience.2016.7870899
- [3] D. De Roure, P. Willcox and A. Abdul-Rahman. "On the description of process in digital scholarship," Proceedings of the 1st Workshop on Humanities in the Semantic Web, co-located with 13th ESWC Conference 2016 (ESWC 2016). <http://ceur-ws.org/Vol-1608/#paper-10>
- [4] D. De Roure and P. Willcox. "Numbers in places: creative interventions in musical space & time", in Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct - Mobile- HCI '16 (2016), ACM pp. 1059-1063. doi: 10.1145/2957265.2964199
- [5] B. Archer. The Nature of Research Co-Design Jan 95 6-13. 1995.
- [6] B.A. Toole (Ed.). Ada, the enchantress of numbers: a selection from the letters of Lord Byron's daughter and her description of the first computer. Strawberry Press: 1992.
- [7] A. M. Turing. 1950. "Computing Machinery and Intelligence," Mind 49: 433-460.
- [8] M. A. Boden. The Creative Mind: Myths and Mechanisms. Weidenfeld and Nicholson, London, 1990.
- [9] A. Petri-Preis. "Emily Howard's Lovelace Trilogy: a musical homage to a mathematical pioneer," Tempo 67 (265) 28-36, 2013. Doi: 10.1017/S0040298213000442
- [10] E. Howard and D. De Roure. Turning numbers into notes. In Ada Lovelace Symposium 2015 - Celebrating 200 Years of a Computer Visionary (Ada Lovelace Symposium '15). ACM, New York, NY, USA, 2015. doi: <https://doi.org/10.1145/2867731.2867746>
- [11] H. L. F. von Helmholtz: On the sensations of tone as a physiological basis for the theory of music, chapter XII, 1895.
- [12] L. Euler: Tentamen novae theoriae musicae ex certissimis harmoniae principiis dilucide expositae, Petropoli, ex typographia Academiae scientiarum, 1739.
- [13] C. Babbage. Passages from the Life of a Philosopher. New Brunswick: Rutgers University Press, 1864.
- [14] R. Duckworth and F. Stedman. Tintinnalogia, or, the Art of Ringing. 1668.
- [15] M. Kallionpää, C. Greenhalgh, A. Hazzard, D.M. Weigl, K.R. Page. and S. Benford. "Composing and realising a gamelike performance for disklavier and electronics," in New Interfaces for Musical Expression (NIME'17), 15-18 May 2017, Copenhagen, Denmark.
- [16] A. Galey and S. Ruecker. "How a prototype argues," Literary and Linguistic Computing, (25) 4, 2010, pp. 405-24. doi: 10.1093/lilc/fqq021