"... there is no reason why it should ever stop": Large-scale algorithmic composition

Nick Collins[1]

[1] Durham University
nick.collins@durham.ac.uk

Abstract. This article surveys some new directions in algorithmic composition, with a special focus on the mass scale of content generation now within reach. Interest in algorithmic composition has recently flourished, including a fashion for deep learning, and commercial offshoots in apps and browser software. Whilst there is much to be excited about in recent developments, we critically survey some current directions, and likely future initiatives, of the field. In particular, the influence of and potential within music information retrieval research is highlighted, corresponding to attempts to bridge between audio machine listening and large corpus training for algorithmic composition. The scenario of mass generation is further considered, including a practical experiment in creating a billion melody data set with accompanying source code.

Keywords: Algorithmic composition, machine listening, state space, perceptual space, data set

1 Introduction

> we set out to prove that if human beings could write 'popular music' of poor quality at the rate of a song an hour, we could write it just as bad with a computing machine but faster (Klein 1957, p. 36).

Algorithmic composition is a well established way for composers to spend their time, though how worthwhile a pursuit is an interesting discussion. A variety of reasons motivate those exploring computer composition, from novel aesthetic ends to historical stylistic analysis and psychological modelling of the act of composition (Pearce *et al.*, 2002). A relatively recent and well referenced survey of algorithmic composition by Fernández and Vico (2013) might make it appear that all that is required for computer-generated music is an of-the-moment machine learning algorithm, and the current fad for deep learning unfortunately reinforces this impression (Briot *et al.*, 2017). Nonetheless, musical decisions rest at the heart of the computational modelling of music creation. This article

takes a critical perspective on the state of play, with the hope that we may better illuminate future research directions and the underlying musical consequences of machine-made music. In particular, I consider a future situation where computer-generated music may even overtake or far exceed human production.

If there are still no algorithmically composed top ten hits, or Hollywood movie soundtracks, it is not through a lack of attempts: consider the reputed five realisations and 1956 radio play of *Push Button Bertha* (Klein, 1957), or Pierre Barbaud's 1960s algorithmic film-scoring work (Viel, 2007). Activity in algorithmic composition has seen a recent rise of commercial interest beyond the academic sphere, including initiatives such as Google's *Magenta* project, IBM's *Watson Beats*, *Artificial Intelligence Virtual Artist* (*AIVA*), Brian Eno and Peter Chilvers' iPhone apps, and François Pachet's Sony pop song album. A recent BBC news article emphasized the narrative of the compositional assistant rather than a more bombastic 'replace-the-humans' trope (Marshall, 2018). The *JukeDeck* startup offer automatically composed music aimed at video content creators otherwise hindered by copyright policing. Their competitors include *Melodrive*, who aim to provide an affective algorithmic composition game engine, *Melomics*, an offshoot from the team who created the *Iamus* contemporary art music score generator (Quintana et al. 2013), and *Amper*. For the latter, a blogger with early access created a song to an *Amper*-generated backing, which accumulated one and a half million views in four months (Southern, 2017). A recent report from the BPI acknowledged the explosive potential of artificial intelligence within the music industry (Music Ally, 2016). Yet adaptive music in video games and generative phone apps has already made clear the mass potential.

I proceed to consider a number of aspects of mass algorithmic composition for the masses. The article has at times the substance of an opinion piece and cannot help but flirt with futurology, though I try to contextualise with respect to academic sources. A number of concrete research suggestions are made, and a trial experiment to produce a billion algorithmic composition outputs is undertaken, with sample code provided in appendices.

2 Why Algorithmically Compose Yet More Music?

There are potentially billions of content creators around the planet, working in a diverse ecosystem of historic and currently active musical styles. Given the fecundity[1] of creation on display, uploaded every day *en masse* to *YouTube*, *SoundCloud* and other sharing sites, why would a generative music program have anything more to contribute? As Mute Records' founder Daniel Miller has

---

[1] Would the AI version be fecundIT?

written, '[a]s people already have access to so much music, generative works only add to the problem of having too many choices' (in Collins & d'Escriván, 2017, pp. 292–293).

Mobile phone apps and video games are areas where generative music programs have gained some traction, but the demands of these media towards interactivity, which may well enable worthwhile musical encounters, is not the pure generativity of a computer program unhampered by human cognition times. Generative music programs without interactivity can generate works substantially faster than the time it takes to listen to them. An algorithmic composer might be tempted to flood the market, and scheme to mass produce music; Thor Magnusson has created a generative music add-on for the *Weavr* digital information agents (Solon, 2012), where they can maintain their own *SoundCloud* account and listening community. But the Internet's early days of anarchic equality have long since been overthrown by conventional limited mass-media channels, and even a viral project is up against digital humanity's short attention span. This is not to say that the promotion of an AI could not potentially have some success; there have in fact been viral AI remixing projects, such as *DonkDJ* or the *Infinite Jukebox* (Lamere, 2012).

Music is an essential human phenomenon, so one answer to the question of value is that the creation of precisely defined compositional theories, as absolutely embodied in program code, provides a window into the human creative process, attempting to make explicit what is otherwise implicit. There is great potential in automations of the creative process, from understanding and preserving personal musical style, to corpus formation for training machine listening algorithms (David Collins, 2016; Cope, 2005; Deliège & Wiggins, 2006).

To better illuminate the possibilities (and generative music systems tend to be all about possibilities), I simplify the four motivations outlined by Pearce and colleagues (2002) to three, removing the 'creation of a tool' category as not directly connected to the status of the algorithmic composition itself,[2] to encompass:

1. Novel compositional work: Computational process can itself be inherently inspiring, and new routes to compose can reveal new output possibilities. Computer generation is especially helpful where heavy calculation is an inherent part of preparing a composition, as with new data sources and new mappings, or following through the implications of newly devised musical rule sets (Doornbusch, 2002).

---

[2] A problem with the Pearce *et al.* (2002) taxonomy is, why not have additional categories for the creation of tools for historical musicology or cognitive modelling, as well?

2. Stylistic analysis and composition: Acuter understanding of historical musical style, and the evolution and mimetics of musical style, is accessible through computational modelling. The computer enforces absolutely rigorous specification of compositional rules to the benefit of reproducible and clear research, and allows artificial simulations of creative spaces to be run that could never otherwise be undertaken (Miranda & Biles, 2007; Cope, 2001)

3. Modelling of the act of human composition: This is an avenue within music psychology research, and may inform attempts to engineer computational creativity based on research into human compositional activity (Cope, 2005; David Collins, 2016).

This separation is by discipline, namely composition, musicology, and cognitive science. These boundaries can still blur on closer examination, for example, due to the unavoidable nature of musical influence within the creation of new music, and in the analysis of historical composition with respect to memes necessarily entering into consideration of compositional process.

Cutting across all three disciplines is the growing capacity seen for music AIs as creators on a mass scale. Algorithmic composition AIs enable mass production of music impossible to an individual human composer; the modelling of the act of creation, historical or cognitive, necessitates engagement with the mass field of musical examples. Modelling an individual human creator may also capture human creativity in a form able to outlive the human originator, another order again of meta-composition again from the composition of a program to create compositions (Taube, 2004). There are heady commercial possibilities (e.g., license-free unique music on tap for content creators, à la *JukeDeck*), and there is the potential to create inexhaustible ground truth data for MIR system development if the algorithm and the perceptual quality of its outputs can be trusted (Sturm & Collins, 2014).

For any of these, humans are always involved, both in writing the algorithmic composition programs, and in assessing the results. It is possible to predict a future AI ecosystem of machines subsisting on each others' compositions, where the standards of listening have significantly deviated from any human listening benchmark; but this is far enough from worthwhile actuality as to be of only current interest to speculative art projects (for instance, a project to install a self-contained environmentally reactive sensor network in the Mojave desert; see Howse, 2007). Arthur C. Clarke's statement might come to mind: "[t]he prospect for modern music is a little more favourable; now that electronic computers have been taught to compose it, we may confidently expect that before long some of them will learn to enjoy it, thus saving us the trouble" (Clarke, 1999). However, we continue forwards on the assumption that human involvement with algorithmic composition research will remain essential, and that

Clarke's invective misses the deeper research outcomes of inspiring and understanding human music making through machine adjoints.

3 Unending Criticism

What would it mean to say that the problem of algorithmic composition was solved?[3] Despite human efforts to seek greater refinement of compositional technique, there is no perfect piece, and the creation of music is a perpetual search. Indeed, against the changing judgement of a complex society, no consensus on perfection could ever be reached, even at a single instant of time. This principle would also hold even for an individual, themselves transforming as they age, with their innate human curiosity always holding them back from declaring any one musical product the last ever required.[4] If it is too exhausting to seek perfection, why not let a machine worry about that for you? The unachievable asymptote within human judgement seemingly contrasts with the quickly programmed closure of a simple-minded computer program; but as the AI attempts to take on more self-critical aesthetic weight, we anticipate the future intensive compute cycles of creative machines striving, struggling, to produce better works. The question of what really motivates an AI, and how it would prove it holds such motivation, is worth asking, since creative engagement is predicated upon acting appropriately to promote itself within existing human cultural society, as per the demands of Colton *et al.* (2011).

Many algorithmic composition systems receive little to no evaluation from third-party listeners, theorists and composers (Pearce *et al.*, 2002). Even if there has been increased awareness of the potential for expanded system evaluation efforts since the millennium, there has not been a significant step-change in the quantity and quality of evaluation experiments undertaken. Such experiments are problematic for many researchers because of a need to take on issues of experimental design and psychological testing, with associated costs in researcher time. Sampling online responses from unknown respondents, bringing in twenty undergraduate students to a lab, or gathering a small pool of expert judges for qualitative feedback, are not necessarily strong guarantees of the efficacy of a system. Even the knowledge that music is computer-generated is enough to instigate bias (Moffat & Kelly, 2006; Cope, 2001), and the standard scientific paradigm in psychology research is acknowledged as having issues

---

[3] This paraphrases the close of David Lodge's novel *Small Worlds* (1984), where an English literature academic asks a conference panel what it would mean for the discipline if they all agreed with each other.

[4] Aesthetics is always an incomplete discourse, because it must reflect the gaping lack of an ultimate arbiter; yet it simply reflects the uncertainty of life.

(Open Science Collaboration, 2015). Given reduced funding for computer music with respect to mainstream psychology, when would results ever be reproduced? There is no great incentive for computer music researchers to test each others' systems further, when there are so many interesting creative goals and divergent historical musics to explore. Instead, a productive avenue pointed to by some trends in recent algorithmic composition is to seek real-world musical testing, through interaction with composers, performers and audiences (Sturm and Ben-Tal, 2017; Collins, 2012). We can trust that if musical outputs are inadequate, they will be quickly mocked by encultured listeners.

The subjective domain of music, built for eternally fluid disagreement that oils societal bonds, admits no easily available single evaluation metric (Loughran and O'Neill, 2017; see also Ian Cross' notion of 'floating intentionality', Cross 2001). Yet more highly developed artificial musicians seeking autonomous behaviour require self-critical facility (Rowe, 1993; Galanter, 2012), and thus the question of evaluation and a researcher's working answer sits at the core of the systems themselves. Even if deferred to a function arising from a machine-learning process over a corpus of training examples, hard decisions on the constitution of that corpus and the representation within which machine learning can operate were already made.[5] We might speculate on the appearance of a future system for transformative creativity (Boden, 2003), equipped with a suite of machine learning algorithms, and the ability to track human social trends in music consumption and to obtain music examples to match, somehow initiating its own constantly re-evaluated growth; but there would remain meta-parameters, including the terms of engagement with society.

Further, critical consensus at a given moment is not eternal judgement. Many composers, currently much overlooked, hope that attitudes towards them will change, perhaps after their death, though hopefully sufficiently before for them to acquire a few redeeming commissions. Unfortunately, the reputation of putative great composers is often well established in their lifetimes, and vested musicological coverage may frighten away a deeper critique of the overwhelming historicity of these lucky figures. The vast number of people we see now aspiring to the status of composer is fair warning that future creators will have no time to pore over discarded figures of the past, but will be too busy trying to establish their own grasp on perpetuity against the flood of musical works.

---

[5] Whilst much is achievable by rule-based approaches, the move away from knowledge-based AI and the drive towards big data has seen increasing deployment of machine learning algorithms working over corpora of example material. The danger is that knowledge explicitly encoded in rules by musician-programmers, mirroring the efforts over centuries of music theorists to develop systematic description of musical composition, is sidelined. Instead, such knowledge becomes implicit or a sidenote of the representational assumptions underlying the capacity for corpus analysis.

If measures of critical success are sought, the most that can be expected is relative success, such as winning a competition by being better than the other entries in the opinion of the jury, or higher performance on certain real real-world socioeconomic metrics (such as sales, number of listens, etc.).

There is a case worth examining where numbers are assigned for success, namely university (or school) marking of composition. Yet no third-year undergraduate can attain 100% in a techniques portfolio of historic composition or in a portfolio of new music free composition. There are always critical bases from which examiners see, if not obvious infelicities of stylistic expectation, at least undeveloped potential. If humans cannot get 100%, what hope for the machine composer?

Consider this extract from the marking criteria for a historical techniques composition portfolio (Durham university music department Level 3 undergraduate module, top band 86–100):

> Exemplary: This work evinces an exceptional level of technical competence and complete assurance in composing within the stylistic parameters prescribed, as well as remarkable imaginative flair and ingenuity. It suggests a profound and extensive familiarity with the musical literature of the historical period(s) studied. Presentation is immaculate.

Note that whilst the parameters of success initially seem pinned down, the remaining leeway for 'flair and ingenuity' and wriggle room around 'familiarity' across a wide set of musical literature are escapes. A qualitative judgment is made within a band of 15 available marks; the department has never awarded anything near a 90. Consider also this excerpt from the free composition assessment criteria for the same level:

> Work of exceptional quality, demonstrating a remarkable degree of originality and sophistication, exemplary technique, complete assurance in its familiarity with contemporary musical culture, and the potential to communicate very powerfully in performance. The work represents a genuine contribution to contemporary art music. Presentation and notation is [sic] exemplary. The written commentary is to a standard potentially worthy of publication in an academic journal, demonstrating exceptional competence and initiative for an undergraduate student.

The examiners will tend to agree if pieces (and associated commentaries, another source of wriggle) are high quality, though again no mark of 90 or above has ever been achieved.[6]

The basis for panel judgements in music competitions are rarely discussed or released, and much feedback to the professional composer is indirect, in the substance of further commissions, rare repeat performances, and the like. Nonetheless, algorithmic composition programs eventually need to compete in real-world circumstances, though they may need some undergraduate study first. It will help algorithmic composition researchers more thoroughly to interrogate composition teachers, competition judges, and the general public, as to how they assign value, and consistently recognise quality.

Assuming that measuring the success of generated music is at all feasible, at least to some level of critical consensus, what proportion of a program's outputs should be brilliant music? Human creators have fallow spots and writer's block, can fall into repetition of well established tropes from earlier in their career, or push so far on some strange new avenue that they fail to bring any audience with them. Why should MusAIcians be any different? Would not the most human thing be to make mistakes, to learn from failure and to overcome missteps only with further works? The danger is, as ever with human-emulating systems, to anthropomorphise too quickly, and to build in hard-coded mistakes whilst claiming some profundity of life experience within the machine.

There is nonetheless a tendency to think that any program that does not consistently make its author proud of it is wayward and programmed poorly. How can the interesting aesthetic missteps of a richly complex system be differentiated from simple limitations of conception? The Turing test rests on human language ability (Ariza, 2009), and a human composer could defend and explain supposed errors of judgement; this disconnection of computer calculations from social embedding is an oft-recurring theme (Colton *et al.*, 2011).

The more autonomous the algorithmic composer grows from its human programmer, the more considerations of machine ethics come to the fore. Is one bad error of judgement enough to spoil the reputation of a computer as to all future output? How long does it take human musicians to be forgiven a musical mistake? How does a human musician re-invent themselves? Should a critical mauling of a new algorithmic composition program halt future development? Currently, the human overseers of such programs make those decisions on the programs' behalf, but increased autonomy of MusAIcians will bring up interesting conundrums mirroring the up-and-down creative life of human musicians.

---

[6] There is a pressure to mark to the curve of standard results in university degree outcomes, but the principle of investigating why humans cannot achieve perfect marks remains.

4 Listening to Everything

A mathematically enormous state space is unavoidable for any non-trivial algorithmic composition system, because of music's combinatorial explosion (Prokofiev, 1978). A corollary is the difficulty of tracing the possible program routes, critical for real program understanding, including debugging (Collins, 2008). Although the perceptual space of the range of musical outputs may be much smaller than the mathematical space (Collins, 2008), it is still potentially vast, if also potentially observer-dependent. At the coarsest scale, an automatic drum and bass generating program probably creates only drum and bass works (though a truly creative system, arguably, should be ready to deviate from such a pattern as inspiration strikes), and could be dismissed by a human antagonistic to the genre as always sounding the same. Only invested experts can really test discrimination at finer scales, and here there is a close connection between listener pattern-spotting, as they try to 'see through' how an algorithm works, and the demands of style as restrictions on plausible patterns.

Calculations of the size of mathematical or perceptual space depend upon the model. As a brief example with perceptually motivated values, consider restricting a monophonic melody generator to short-term memory (under 36 notes; see Snyder, 2000), a melody within a limited compass, say, one octave of a diatonic scale (easily within 7+-2), and two rhythmic values (à la music psychologist Paul Fraisse's fast and slow durations; see Fraisse, 1982; Clarke, 1999). A 36-note melody so construed has $7*14^{35}$ possibilities. Order 10^{40} possible melodies is still rather big. In practice, the 36 notes are too many to follow with reliable comparative judgement, but because real ecologically valid music will have multiple parts set within the additional complicating factors of timbre and space, even with co-conditioning harmonic and rhythmic structure, the state space remains larger than any individual human's available lifetime of listening.[7] Even a between-subjects (rather than a within-subjects) experimental design offers

---

[7] Assume a human lives 80 years, and their developed adult judgement is active for 62 of those. This corresponds to $(62*365+16)*24*60*60 = 1,956,614,400$ seconds, that is, two billion seconds (or one billion if they are allowed to sleep and to have a little other time off). If one of the perceptual melodies is under 10 seconds, again for short-term memory requirements, the number an individual can audition is around $2*10^8$, which is tiny compared with the actual state space. If, in a rather speculative future, the human lifetime is seriously extended, or the 'problem' of death overcome with brain backups (O'Connell, 2017), the transhuman composer may produce quite a few more opuses than is usual today, before the heat death of the universe. 'Terminal escape velocity' would still not be fast enough to compete with the rate of machine composition, however, at least without the sort of cyborg upgrades that bring the computer programmer rather closer to their program.

little help, since recruiting all of humanity only improves matters by a multiplier of 7 billion.

How does anyone debug a big-budget computer game? Programmers must content themselves to remove the most egregious gotchas and hope that the sample of playing time carried out by their testers is sufficient to catch typical bugs. It is often remarked in software engineering that all production release software has many bugs left in, and that any software update might plug some holes but continues to provide new avenues for program error (Myers *et al.*, 2012). We might stretch the analogy to composers: as composers develop within their lifespan, they introduce new interesting quirks even as they aesthetically work out old mannerisms, and this sort of rich territory is essential to any developed composition engine engaging with culture.

The enormous scope of pre-existing music sets up further issues. Since no human is expert in all the world's musics, why should any single MusAIcian be? Although music psychologists have posited possible universals of human musical cognition (Stevens & Byron, 2009), these tend to be quite low-level psychoacoustic attributes of the human auditory system rather than culturally contingent top-level musical particulars (Patel & Demorest, 2013); music is a cognitive cultural iceberg. Evolution has furrowed out human universal constraints in the basic processing of sound, and society imposes particular treatment of musical education and the musics of importance to that culture, such that we would always expect some axioms and preconditions of musical training. Yet the interconnection of human cognition with real-life complexity, and the non-isolation of musical sub-modules, brings another case of a supposedly domain-limited problem, musical creation, expanding on closer investigation to become the whole hard-AI question.

Ethnomusicologists are highly sensitive to claims of musical universals that might undermine cultural uniqueness (Miller & Shahriari, 2009; Feld, 1974). With the grounding of music in cultural conventions and histories, clashes of political, cultural and belief systems between peoples may affect the ability of any one individual to appreciate 'all' music. If the basis underlying musical conventions are in contradiction, we cannot expect any consistent rational universal AI to understand all musics in a deep sense. The musical surface of audio recordings is always amenable to automated listening analysis, but the programmer who sets up the machine-listening situation will be pre-establishing representational bias, and further complicating this by importing support libraries with their own associated representational assumptions in turn.

Against so many musics and so many meanings for music (Cross and Tolbert, 2009), generalizability cannot be expected to carry well across the whole music domain. Instead, algorithmic composition projects tend to be highly constrained to a particular target musical style. In this context it is certainly worth investigating freer composition systems that are inspired by a wider range of music

than that of the musical style directly under generation, since musical hybridity is such an important part of compositional exploration.

Influence is ubiquitous; creators are always operating with respect to their implicit knowledge of the past, even if we might stop short of calling them highly biased plagiarists. They may not consciously admit influences, but formative influence always lurks and guides compositional choice.[8] Corpus-based composition systems have an advantage in that they make absolutely explicit what influences they require. However, study of musicians' attitudes to influence has shown that humans are not overly anxious about influence (Collins, 2011). A richly creative AI may need to forget the circumstances of its learning about a particular training example, so as not to be overly burdened by the weight of history.

5 The Interconnection with Machine Listening

Machine listening is the modelling of human auditory faculties by computer. Few algorithmic composition projects enclose the act of machine listening, though listening is critical to human creation and criticism (Collins, 2012). The conflux of machine listening and algorithmic composition has great potential as a research area, and is the remedy to Daphne Oram's warning on mass production of machine music as also appearing in this article's title:

> complex music "by the yard" would be produced. Hours and hours of music, days and days of it ... there is no reason why it should ever stop! It will be as defined or as aleatoric as your program has decreed. Unlike the human performer, the computer has no natural rationalising department. It will not try to "make sense" of the sounds it is producing, nor will it ever intuitively know when the poor listeners have had enough! (Oram 1972, p. 87).

Machine listening is the general step to model a real human's encounters and development with musical history. The audio corpus studies at the heart of the Music Information Retrieval (MIR) research agenda have strong potential as databases for training machine listening systems. We can envisage a future feedback loop (Fig. 1), where output algorithmic compositions are created by systems trained on real musical examples, and algorithmic outputs may in turn

---

[8] Copyright law often has nonsensical assumptions at its base: what if (limited, model-based) plagiarism is crucial for any creative work whatsoever?

become the next generation of available music.[9] Mass rendering, and mass listening, may become an essential tool, even for next-generation composers, who can expand their musical awareness beyond the music they ever had time to listen to, to take in machine models founded in much larger datasets (Collins, 2016).
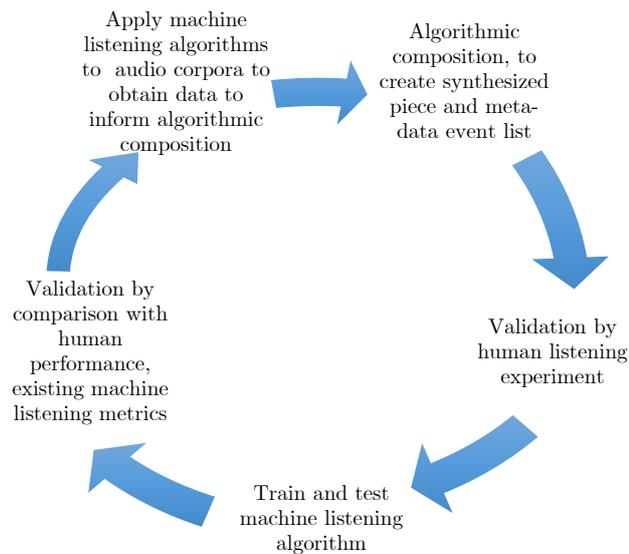
Apply machine listening algorithms to audio corpora to obtain data to inform algorithmic composition

Algorithmic composition, to create synthesized piece and metadata event list

Validation by human listening experiment

Train and test machine listening algorithm

Validation by comparison with human performance, existing machine listening metrics

Figure 1: Bootstrapping loop combining algorithmic composition and machine listening research.

The musical AI has the potential to listen to and learn from more music than any human being can in their lifetime, because audio files are now commonly analysed faster than real-time, and listening can be parallelised across a cluster of computers. The caveat is on the quality of machine 'listening' in terms of emulating human cognition.

Human listeners have ingrained preferences and prejudices (North & Hargreaves, 2008). Should MusAIcians have favourite composers and pieces? What does it mean for an AI to listen to an audio file again and again, in the same way that a human fan of a recording artist might listen on repeat to their latest release? Would the notion of strongly liking a piece of music in any deeper sense,

---

[9] Autonomously generating music may allow stepping up a gear in MIR research, by providing vast quantities of fully annotated music after a bootstrapping stage from ecologically valid starting examples.

adapting your life to accommodate that music more centrally, necessitate cutting off appreciation of music far from it? There is a much repeated claim by humans that they have wide ranging tastes and a very diverse record collection, but this is often a severe exaggeration when set against the sheer mass of music available around the world.[10] What level of proven machine listening ability will it take for human musicians finally to defer to the judgement of MusAIcians who have such vastly greater direct experience of the mass of music in culture?

6 Billion Song Datasets

The Million Song Dataset is a well known resource in MIR, including audio feature data for one million popular music pieces with a bias to 1990s and 2000s releases (Bertin-Mahieux *et al.*, 2011). Estimates of the amount of music in online providers places it at around 30–65 million pieces from *Spotify* to *iTunes*. However, as a case study in algorithmic fecundity, we could aim to make more music than recorded in human history by creating a billion song dataset.[11] This is not an entirely academic pursuit, but potentially transformative in the order of music calculated by machine, pointing the way to musical societies where music AI predominates.

If we assume much faster than real-time generation, at one second of computer time to create a single output piece, it takes 31 years to create one billion pieces. Pricing for cloud computation on a single core might cost $0.1 per hour, working out over that time span to only $27,156. Practical generation requires parallelisation over multiple cores in a cloud-computing cluster. If we assume 1,024 cores at once, it could only take a third of a year!

For a more practical fast-paced generation, it is clear we need to get the generation time much shorter than the second; we could as a first step avoid audio rendering, and only work in the symbolic domain. We now work through a test example, to assess the feasibility of such an operation. Appendix A gives a *SuperCollider* code implementation of the flow diagram for the classic algorithmic composition project *Push Button Bertha*'s melody generation (Klein, 1957). Since the source text does not give sufficient detail on the rhythm choices, a simple model has been deployed here. Benchmarked in *SuperCollider*, around 10,000 36-note melodies can be created per second, or one billion 'songs' in 26 hours of compute time. To push this faster, we recode the example in Fortran, the fastest non-assembler option, with code provided in Appendix B. One billion melodies are now generated in just over one hour (61m 42s); this was run on

---

[10] MIR offers the capability actually to test this diversity of collection.

[11] This dataset may never exist on disk as one entity, but could be a resource within online training of a model.

Tuesday 6 March 2018, at lunchtime.[12] This does not, however, outstrip all human melody writing to date, since the number of human beings ever alive has been estimated at around 100 billion (Haub, 2011), and we might want to credit almost everyone with 'writing' a unique melody at some time in their lives, professional composers aside,[13] if only absent-mindedly whistling out of tune. If we make the gross assumption of one new melody per individual, the Fortran program could exceed human history in under five days.

Churning through a combinatorial tract might seem cheating, however, since the songs only exist as platonic entities. Perhaps basic signal manipulations of existing song fragments could help speed up generation in the audio file case, such that the layering between and ordering within parts is shuffled to create many subsidiary pieces for each long generated work; otherwise, the rampant parallelism available through cloud computing promises new mass capacity.[14]

Video games or music apps selling millions of copies provide a mechanism where billion 'song' datasets have already arisen. If a video game sells 25 million units, only 40 rendered scenes/levels requiring distinctly generated music are necessary per player before one billion distinct generative music excerpts have been created over all the gamers' computers.[15] If future human society spreads beyond one planet or solar system and hits trillions, and only one billion make their own generative music programs, or some percentage are working with existing musAIcian programs, the level of music generation again hits new heights. We might anticipate, beyond Mahler's 'Symphony of a Thousand', a symphony of one billion AIs.

7 Conclusions

Aspects of mass algorithmic composition[16] have been explored in the hope that the reader may be provoked to reconsider some assumptions they may have had

---

[12] The computer used was a five-year-old MacBook Pro, with only one core. Multi-core processing with a more recent machine might approach an order of magnitude increase.

[13] There is absolutely no claim here that all historical human societies had any notion of the role of a composer, nor that their music was at all similar to the simple diatonic melody model of *Push Button Bertha*, though the basic existence of music making in all societies is less controversial (Cross, 2001).

[14] Or, more illegally, through zombie networks of infected machines. The composer who would write virus programs to invade the Internet of Things for algorithmic musical purposes may already be at work.

[15] If we extend to the generativity of performance, we might claim that *Happy Birthday* has been performed by at least one billion human beings at some point in their lives, and the micro-variations of their delivery constitutes another billion song dataset.

[16] I have tried to avoid composer puns such as Mass-e-net, Datahaven and Corperin so far in this article, but have now lapsed.

concerning the potential and directions of research. I bear in mind the contrast of research intentions from composition to musicology and cognitive science, and point to the potential of machine listening in the algorithmic composition domain. No ultimate solutions are offered for the problems of evaluation; instead, I push for greater acceptance of the divergent definitions and tastes in music that will ultimately make any consensus solution of algorithmic composition meaningless.

Algorithmic composition is already deeply suffused within human musical life, and we cannot unmake its radical potential. From one perspective, the domain is already solved, inasmuch as artificially creative systems are now recursively influencing human musical culture. The remaining permutations of algorithm and evaluation can productively and processively fill the rest of time. If the Lovelace test (Loughran and O'Neill, 2017) requires music generation to confound its own system programmer, the totality of the many algorithmic music systems easily pass the test *en masse*, because they transcend the ability of any one human to comprehend all their details, and machine learning developments surely place single systems on the path to creative autonomy. What a time to be AIive!

References

Ariza, C. (2009). The interrogator as critic: The Turing test and the evaluation of generative music systems. *Computer Music Journal, 33*(2), 1–23.

Bertin-Mahieux, T., Ellis, D.P.W., Whitman, B. & Lamere, P. (2011). The million song dataset. In *Proceedings of the 12ᵗʰ International Society for Music Information Retrieval Conference (ISMIR 2011)*. Miami, FL: University of Miami.

Boden, M. (2003). *The creative mind: Myths and mechanisms*. London: Routledge.

Briot, J-P., Hadjeres, G. & Pachet, F. (2017). Deep learning techniques for music generation: A survey. arXiv preprint arXiv:1709.01620.

Clarke, A.C. (1999). Rocket to the renaissance. In I.T. Macauley (Ed.), *Greetings carbon based bipeds!* London: Voyager.

Clarke, E. (1999). Rhythm and timing in music. In D. Deutsch, (Ed.), *The psychology of music* (2nd ed., pp. 473–500). San Diego, CA: Academic Press.

Collins, D. (Ed.). (2016). *The act of musical composition: Studies in the creative process*. Burlington, VT: Routledge.

Collins, N. (2008). The analysis of generative music programs. *Organised Sound, 13*(3), 237–248.

Collins, N. (2011). Musicians' attitudes to musical influence. *Empirical Musicology Review, 6*(2), 103–124.

Collins, N. (2012). Automatic composition of electroacoustic art music utilizing machine listening. *Computer Music Journal, 36*(3), 8–23.

Collins, N. (2016). Towards machine musicians who have listened to more music than us: Audio database led algorithmic criticism for automatic composition and live concert systems. *Computers in Entertainment, 14*(3), 1–14.

Collins, N. & d'Escriván, J. (Eds.) (2017). *The Cambridge companion to electronic music* (2nd ed.). Cambridge: Cambridge University Press.

Colton, S., Charnley, J. & Pease, A. (2011). Computational creativity theory: The FACE and IDEA models. In *Proceedings of the Second International Conference on Computational Creativity* (pp. 72–77). Mexico City. Retrieved from http://computationalcreativity.net/iccc2011/proceedings/index.html.

Cross, I. (2001). Music, cognition, culture, and evolution. *Annals of the New York Academy of Sciences, 930*(1), 28–42.

Cross, I. & Tolbert, E. (2009). Music and meaning. In Hallam, S., Cross, I., & Thaut, M. (Eds.) *Oxford handbook of music psychology* (pp. 24–34). New York: Oxford University Press.

Cope, D. (2005). *Computer models of musical creativity*. Cambridge, MA: MIT Press.

Cope, D. (Ed.) (2001). *Virtual music: Computer synthesis of musical style*. Cambridge, MA: MIT Press.

Deliège, I. & Wiggins, G.A. (Eds.) (2006) *Musical creativity: Multidisciplinary research in theory and practice*. Hove: Psychology Press.

Doornbusch, P. (2002). Composers' views on mapping in algorithmic composition. *Organised Sound*, *7*(2), 145–156.

Feld, S. (1974). Linguistic models in ethnomusicology. *Ethnomusicology*, *18*(2), 197–217.

Fernández, J.D. & Vico, F. (2013) AI methods in algorithmic composition: A comprehensive survey. *Journal of Artificial Intelligence Research, 48*, 513–582.

Fraisse, P. (1982). Rhythm and tempo. In D. Deutsch (Ed.), *The psychology of music* (1st ed., pp. 149–180). San Diego, CA: Academic Press.

Galanter, P. (2012). Computational aesthetic evaluation: Past and future. In J. McCormack and M. d'Inverno (Eds.) *Computers and creativity* (pp. 255–293). Berlin: Springer.

Haub, C. (2011). How many people have ever lived on earth? *Population Reference Bureau*, October 2011. Retrieved from http://www.prb.org/Publications/Articles/2002/HowManyPeopleHaveEverLivedonEarth.aspx.

Klein, M.L. (1957). Syncopation by automation. *Radio-Electronics*, June, 36–38.

Lamere, P. (2012). The infinite jukebox. Blog post, November 12, 2012. Retrieved from https://musicmachinery.com/2012/11/12/the-infinite-jukebox/.

Loughran, R., & O'Neill, M. (2017). Limitations from assumptions in generative music evaluation. *Journal of Creative Music Systems*, *2*(1). Retrieved from http://jcms.org.uk/issues/Vol2Issue1/limitations-from-assumptions/article.html.

Marshall, A. (2018). Is music about to have its first AI No.1? *BBC Music*, Wednesday 28th February 2018. Retrieved from https://www.bbc.co.uk/music/articles/0c3dc8f7-4853-4379-b0d5-62175d33d557.

Miller, T.E. & Shahriari, A. (2009). *World music: A global journey* (2nd edition). New York: Routledge.

Miranda, E.R. & Biles, J.A. (Eds.) (2007). *Evolutionary computer music*. London: Springer-Verlag.

Moffat, D. C. & Kelly, M. (2006). An investigation into people's bias against computational creativity in music composition. In *Proceedings of the 3rd International Joint Workshop on Computational Creativity* (pages unnumbered). Italy: Riva del Garda.

Music Ally (2016). Music's smart future: How will artificial intelligence impact the music industry? Report released by the BPI. Currently unavailable online; last accessed 13 December 2016.

Myers, G.J., Badgett, T. & Sandler, C. (2012). *The art of software testing* (3rd ed.). Hoboken, NJ: John Wiley and Sons.

North, A., & Hargreaves, D. (2008). *The social and applied psychology of music*. New York and Oxford: Oxford University Press.

O'Connell, M. (2017). *To be a machine: Adventures among cyborgs, utopians, hackers, and the futurists solving the modest problem of death*. London: Granta.

Open Science Collaboration (2015). Estimating the reproducibility of psychological science. *Science*, *349*(6251), aac4716.

Oram, D. (1972). *An individual note of music, sound and electronics*. London: Galliard Ltd.

Patel, A.D. & Demorest, S.M. (2013). Comparative music cognition: Cross-species and cross-cultural studies. In Deutsch, D. (Ed.), *The psychology of music* (3rd ed., pp. 647–681). London: Academic Press.

Pearce, M., Meredith, D. & Wiggins, G. (2002). Motivations and methodologies for automation of the compositional process. *Musicae Scientiae*, *6*(2), 119–47.

Prokofiev, S. (1978). *Sergei Prokofiev: Material, articles and interviews*. Blok, USSR: V. Progress Publishers.

Quintana, C.S., Arcas, F.M., Molina, D.A., Rodriguez, J.D.F. & Vico, F.J. (2013). Melomics: A case-study of AI in Spain. *AI Magazine*, *34*(3), 99–103.

Rowe, R. (1993). *Interactive music systems*. Cambridge, MA: MIT Press.

Snyder, B. (2000). *Music and memory: An introduction*. Cambridge, MA: MIT Press.

Solon, O. (2012). Weavrs: the autonomous, tweeting blog-bots that feed on social content. *Wired UK*, Wednesday 28 March 2012. Retrieved from http://www.wired.co.uk/article/weavrs-spambots-or-discoverability-agents.

Southern, T. (2017). Break Free, Taryn Southern (Official Music Video). Retrieved from https://www.youtube.com/watch?v=XUs6CznN8pw.

Stevens, C. & Byron, T. (2009). Universals in music processing. In Hallam, S., Cross, I. & Thaut, M. (Eds.) *Oxford handbook of music psychology* (pp. 14–23). New York: Oxford University Press.

Sturm, B. & Ben-Tal, O. (2017). Taking the models back to music practice: Evaluating generative transcription models built using deep learning. *Journal of Creative Music Systems*, *2*(1). Retrieved from http://jcms.org.uk/issues/Vol2Issue1/taking-models-back-to-music-practice/article.html.

Sturm, B. & Collins, N. (2014). The kiki-bouba challenge: Algorithmic composition for content-based MIR research and development. *Proceedings of IS-MIR 2014*. Taipei, Taiwan. Retrieved from http://www.terasoft.com.tw/conf/ismir2014/Proceedings.html.

Taube, H. (2004). *Notes from the metalevel: An introduction to computer composition*. New York, NY: Routledge.

Viel, N. (2007). Pierre Barbaud et la naissance de la musique par ordinateur en France: De la cybernétique à l'algorithmique. *Proceedings of EMS : Electroacoustic Music Studies Network*. Leicester: De Montfort University.

Appendix A

SuperCollider code instantiating the *Push Button Bertha* melody-generation algorithm from Klein (1957).

```
(
var lastnote = [0,1,3,5,6,7,8].choose;
var notes = List[[-1,3],[lastnote,1]];
var nextnote;
var nextdur;
var barpos = 0;
var chooseduration;

chooseduration = {

    nextdur = 10.rand;

    if(nextdur<7)                        {nextdur                =
[1,1,1.5,2,4,0.5,0.5].at(nextdur)}{

            nextdur = 4.0-(barpos%4.0);

    };

    barpos = barpos + nextdur;

    nextdur;

};

//original  creates  50  notes,  but  requiring  short-term
memory constraint here
while({notes.size<36},{

    var test  = true;

    while({test},{

            nextnote = rrand(0,9);

            if(abs(nextnote-lastnote)<6) {

                    test = false;
            };

    });

    notes.add([nextnote,chooseduration.()]);
```

```
    if(nextnote==0)      {nextnote     =     rrand(1,2);
notes.add([nextnote,chooseduration.()]);};
    if(nextnote==9)      {nextnote     =     5;
notes.add([5,chooseduration.()])};

    lastnote = nextnote;

});

~notes = notes.collect{|note|

    if( (note[0]) != (-1)) {
        //Bb rather than B natural here

    [[1,0,2,4,5, 7,9,10,12,6].at(note[0])+60,note[1]]
    } {
    //rest
    [\,note[1]]
    };

}.postcs;

)
```

Appendix B

Fortran code instantiating the *Push Button Bertha* melody-generation algorithm from Klein (1957).

```fortran
!gfortran -ffree-form fortranprison.f
real function chooseduration(durationchoices,barpos)
      real nextdur
      real, dimension(7), intent(in) :: durationchoices
      real barpos
      integer choice

      call random_number(r)

      choice = int(r * 10)+1

      if (choice .LT. 8) then
            nextdur = durationchoices(choice)
      else
            nextdur = 4.0 - modulo(barpos, 4.0)
      endif

      barpos = barpos + nextdur;

      chooseduration = nextdur

      return
end


program pushbuttonbertha

integer lastnotechoices(7)
real durationchoices(7)
integer lastnote
!integer note
integer nextnote
real barpos
integer choice
integer noteschosen
integer notepitches(36)
real notedurations(36)
integer pitchnow
integer finalpitch(10)
integer test

real(8) :: r
real chooseduration
```

```
integer :: values(1:8), k
integer, dimension(:), allocatable :: seed

call date_and_time(values=values)

call random_seed(size=k)
allocate(seed(1:k))
seed(:) = values(8)
call random_seed(put=seed)

finalpitch (1:10) = (/ 1,0,2,4,5,7,9,10,12,6 /)
durationchoices (1:7) = (/ 1.0,1.,1.5,2.,4.,0.5,0.5 /)
lastnotechoices(1:7) = (/ 0,1,3,5,6,7,8 /)

!one billion generations
do 77 M=1,1000

print *,M*1000000

do 777 L=1,1000
do 7777 K=1,1000


call random_number(r)

choice = int(r * 6.999999999)+1
lastnote = lastnotechoices(choice)

notepitches(1) = -1
notedurations(1) = 3
notepitches(2) = lastnote
notedurations(2) = 1
noteschosen = 2
barpos = 0.0

test = 1

do while ( noteschosen .LT. 36 )
            test = 1

            do while (test .EQ. 1)

                call random_number(r)

                    nextnote = int(r * 9.99999999)
```

```fortran
                           if (abs(nextnote-lastnote) .LT. 6)
test = 0

                end do


                    noteschosen = noteschosen + 1

                notepitches(noteschosen) = nextnote
                notedurations(noteschosen)                   =
chooseduration(durationchoices,barpos)
                barpos = barpos + notedurations(noteschosen)

        if ( noteschosen .LT. 36 ) then

             if (nextnote .EQ. 0) then
                 noteschosen = noteschosen + 1
                 nextnote = 1;
                 call random_number(r)
                 if (r .LT. 0.5) nextnote = 2;
                 notepitches(noteschosen) = nextnote
             notedurations(noteschosen)                     =
chooseduration(durationchoices,barpos)
             barpos = barpos + notedurations(noteschosen)
             endif

             if (nextnote .EQ. 9) then
                 noteschosen = noteschosen + 1
                 notepitches(noteschosen) = 5
             notedurations(noteschosen)                     =
chooseduration(durationchoices,barpos)
             barpos = barpos + notedurations(noteschosen)
             endif

        endif
                      lastnote = nextnote;

         end do


do 36 J=1,36

     pitchnow =  notepitches(J)

!Bb rather than B natural here
     if (pitchnow .NE.  -1)   notepitches(J) = final-
pitch(pitchnow)+60
```

```
      !print *, pitchnow

36 continue

7777 continue
777 continue
77 continue

      !array of notes now calculated

end program
```