TWO PERSPECTIVES ON REBOOTING COMPUTER MUSIC EDUCATION: COMPOSITION AND COMPUTER SCIENCE

Ben Swift
Research School of Computer Science
The Australian National University
ben.swift@anu.edu.au

Charles Patrick Martin
Research School of Computer Science
The Australian National University
charles.martin@anu.edu.au

Alexander Hunter
School of Music
The Australian National University
alexander.hunter@anu.edu.au

1. ABSTRACT

Laptop ensembles and orchestras, in addition to being hubs for collectives of experimental musicians, have become a popular feature in music technology tertiary education curricula. The (short) history of such groups reveals tensions in what these groups are for, and where they fit within their enfolding institutions. Are the members programmers, composers, or performers? Should laptop ensemble courses focus on performance practice, composition, or digital synthesis? Should they be anarchic or hierarchical? Eschewing potential answers, we instead pose a new question: what happens when computer science students and music students are jumbled together in the same group? In this paper we discuss what a laptop ensemble might have to offer both groups or students and describe the results of our experiments in running joint laptop ensemble courses. We conclude with questions that motivate further iterations of our laptop ensemble as a promising site of computer music education in a modern university environment.

2. INTRODUCTION

This paper provides two perspectives on encouraging the development of an electronic and experimental music scene among university students. The first perspective is one of composition students within a music department, and the second is of "creative coding" students within a computer science department. Computer music has a long history of collaboration and cross-disciplinary development from music and computing faculties. In our efforts to foster engagement with music from both music and computing students we have combined free improvisation performances and open-work compositions from music with laptop ensembles and software engineering processes from computing. These perspectives embrace commonalities while acknowledging the differences between students on both sides of this interdisciplinary field.

In this paper we report on the activities of *LENS*: the *ANU Laptop ENSemble*. We discuss the tensions between both perspectives, and address the impact of new trends in both music and computing education such as a new focus on broad-audience coding education and music production with the academy. Finally, we evaluate the impacts of these perspectives on the students and examine the computer music practices that have emerged so far.

3. BACKGROUND

While computer music ensembles are hardly a new phenomenon (see Bischoff et al. 1978, for an early example), interest in orchestras of laptops surged in the 2000s at universities in the USA including *PlOrk* (Trueman et al. 2006), *SlOrk* (Wang et al. 2009), *L2Ork* (Bukvic et al. 2010), and others. In contrast to electronic music collectives, the "*Ork" model pioneered in Princeton's PlOrk adapted the hierarchical structure of a university orchestra, being composed of a large number of musicians with identical laptop and speaker setups. Wang et al. discussed how this approach tended to emphasise performance practice with computers and was able to attract those who are less experienced with electronic music (Wang et al. 2008).

The laptop ensemble has proven to be a popular format, with Knotts and Collins identifying a history of at least 160 different groups that could be described as such (Knotts and Collins 2014). The technical and organisational settings for these groups varies wildly; some are essentially piano keyboard ensembles (e.g. Cheng 2019) with centralised sound production, while others are more experimental, such as the *Ensemble Feedback Instruments* described by Rosli et al. where performers freely mix and map the inputs and outputs of DIY electronic instruments (Rosli et al. 2015). Some ensembles focus entirely on rehearsals, others a mix of rehearsal and workshop sessions (Trueman et al. 2006), and still others rehearsal and lectures (Bukvic et al. 2010).

When laptop ensembles exist within universities, what exactly are the students learning? Trueman's original argument for PlOrk was to develop a community of practice within the electroacoustic music studio, which was at risk of becoming an individualised practice (Trueman 2007). In one of the few critical evaluations of the laptop orchestra phenomenon from a participant's perspective, Valiquet notes that students of the *Concordia Laptop Orchestra* (*Clork*) value the technical skills gained over the musical outcome which could be seen to exist in a "kind of detached, tolerant aesthetic pluralism" (Valiquet 2018).

Notably, the most high-profile laptop ensembles have existed within the host university's music programs (either

traditional or music tech) rather than computing/computer science programs. Our wish to meet the needs of both computing and music students in a laptop ensemble appears to be somewhat unique.

4. LENS: THE ANU LAPTOP ENSEMBLE

Over the last few years, we three academics—a live-coder and senior lecturer in Computer Science, a percussionist and computer musician and lecturer in Computer Science, and a composer/improviser and lecturer in Music—have been finding ways to bring our students together in meaningful collaborative multimedia projects. Like the LOrk pioneers, we wish to emphasise student-created works rather than existing repertoire. Perhaps unlike them, we are open to different collaborative configurations with subsets of our enrolled students performing works that might include acoustic instruments and live visuals.

Since early 2015 we have run extra-curricular workshops, summer schools, collaborative projects, installations and site-specific performances. Until 2018, all of these projects have almost exclusively involved music students from the Experimental Music Studio. In part due to recent successes in Swift's *Art & Interaction in New Media* CS course (featuring guest lectures from practitioners across the university) there has been a push from the Computer Science side of campus to reach out and provide more creative outlets and opportunities for students.

Given these recent trends, in 2018 we formalised these previously extracurricular activities in the *LENS*: the *ANU Laptop Ensemble*. In the next section we describe how this has been arranged as a for-credit course at the Australian National University (ANU). However, we first wish to discuss the broader reasons *why* we chose to co-run LENS "across the campus" in this way.

4.1 Why run it as a for-credit course at all?

Since 2018 the LENS course has been offered as a for-credit elective for students from either the ANU Research School of Computer Science or the ANU School of Music. This means that students from either CS or music can take the course for credit, and also that the course counts a degree program elective, not a cross-campus elective.

Running the LENS for university credit raises the question: "if the goal is to create a community, is the best approach to give a mark and a grade for participating?" It is important to remember that today's students are under considerable time and economic pressures, which have been shown to cut into the time which once may have been used for extracurricular activities (Beiter et al. 2015). By offering the course for credit, we both free up and incentivise students to create the time and space in their schedules required to participate fully and achieve good results. Additionally, the extrinsic motivation of having assessment can help bring out the best work in (some) students.

There are challenges with this approach: modern universities are large institutions, and attempting to organise enrolments, teaching spaces, and tutors across departments can be challenging. However, we have managed to leverage our existing (good) relationships between the CS and Music schools to make this happen.

4.2 Why should CS students join LENS?

For some students, being asked to make something creative or artistic is what it takes to move them beyond simply "grubbing for grades" to asking questions that make them better computer scientists and software engineers. LENS also provides a context to encourage the dreamers, doodlers and creatives who choose to study CS at university. Too often, students feel that studying CS means giving up their artistic and creative dreams, and CS schools miss out on some brilliant students because of this (see, for example, the literature on STEAM education (Harris and de Bruin 2017)). Running LENS as a CS course is a way of letting these students know that creativity is important and that it's part of doing top-class engineering, CS and design work.

Computing (in the broad sense) is no longer seen in society as an unalloyed (if boring) force for productivity, labour-saving and the general good. New narratives range from techno-utopian visions of post-scarcity to dystopian nightmares of surveillance and oppression. In this context, university CS departments are wrestling with how to teach students to be thoughtful and ethical as well as skilful and effective.

LENS may seem like a strange diversion in light of these much bigger challenges, but creative computing provides an important test-bed for applying computational tools in domains where there is no "right answer" and no objective measure of quality. LENS forces CS students to build something to a specification, but to decide on what "quality and correctness" looks like before attempting to produce it. This is an important skill in the wider context of computers and society, and LENS gives students a taste of these challenges.

4.3 Why should music students join LENS?

Since 2013 the ANU School of Music has been placing an increasing amount of energy and resources behind the development of programs in music technology and composition for film and video games. As these program grow and students complete existing curriculum, a need for additional avenues within the university to stretch their technological capabilities has arisen.

Participating in LENS encourages music students to create their own electronic music tools rather than rely on pre-made loops or plugins. Music students who may or may not already be familiar with synthesisers and electronic music (most often through Ableton Live or other DAWs) learn the fundamentals of synthesis, as well as taking serial techniques and applying them to computerised algorithmic processes. The composition and performance of music without the familiar interfaces of acoustic instruments, standard notation and notation and DAW software give these students a fresh perspective, and a common entry point with their computer science colleagues.

As noted above, the laptop ensemble puts electronic music into a social and collaborative environment. Those who tend towards electronic music composition and production are challenged to find a way for a group to "play" their composition. The question of "who turns which knob, and when?" brings intuitive practice into deliberate composition.

Figure 1. Ben Drury, Charles Martin, Millie Watson, and Ben Harb performing on touchscreen instruments, 2016.

5. LENS COURSE FORMAT

5.1 Version 0: workshops, projects, and collectives

Our earliest experiments with computer music ensembles were related to individual research projects by the authors. Ben Swift's *Viscotheque* system was trialled by ensembles from 2010–2012 (Swift 2013). Charles Martin founded *Ensemble Metatone* in 2013 to explore the combination of percussion and touchscreen music apps (Martin 2017). Alexander Hunter founded the Canberra Experimental Music Studio in 2014 where music and other students were able to participate in free-improvisation and electronic music performances and recordings (e.g., Hunter et al. 2018).

While these projects were fulfilling, the participants performed with either fully-formed musical systems (as in the case of Swift and Martin's groups) or relied heavily on individual participants' independent learning (as in Canberra EMS). Apart from this, the authors were independently teaching large core computer science and music composition subjects, and encountering students who wanted to make computer music but didn't know how. The following attempts to create LENS were designed with those students in mind.

5.2 Version 1: TechLauncher

We first ran LENS for-credit with a group of 10 students in 2018. These students were from both Computer Science and Music. This pilot program used an existing group project course called *TechLauncher*, which was primarily designed to provide industry client-based tech entrepreneurship projects for later-year CS students. To fit LENS within this structure we assumed roles (client and tutor) usually filled by industry mentors. Our client brief: to provide the space to explore the creative potential of the technology they use every day—to make, break, mend, hack, learn and unlearn.

This arrangement was very flexible: the students had broad scope to make whatever they wanted, and the established enrolment process was particularly useful for the CS students (many of whom were required to take the *TechLauncher* course anyway). However, in comparison to the other groups within the program the LENS cohort was small, and their work wasn't easily understood by other groups engaging in peer assessment. The course's focus on process rather than product limited our ability to critically engage with the musical outcomes.

Figure 2. LENS developing a performance at Ainslie Arts Centre, 2018.

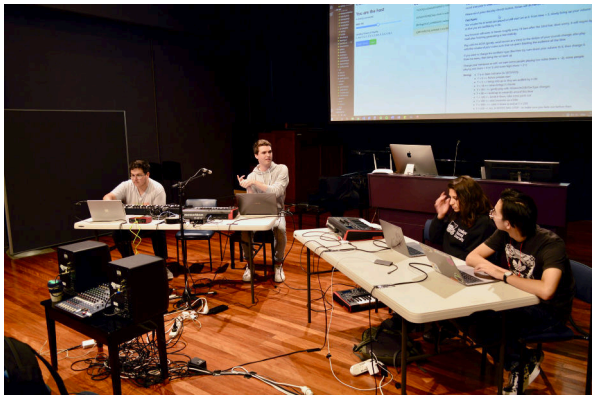

Figure 3. Joshua McConnell, Rohan Proctor, Kira Breithaupt and Weitong Huang rehearsing a LENS work, 2019.

5.3 Version 2: music and CS project

The present iteration of LENS (Semester 2, 2019) similarly reuses existing ANU course codes. Both the CS and music programs already contained open-ended course codes (*Special Topics in Computing* and *Music Project* respectively). Within these course codes, we arranged the LENS enrolment & assessment schedule so that each student developed an individual laptop ensemble project while performing and working together as a group. Both had summative assessment schemes with a 50% weighting on a project report (or process diary) and 50% on the practical outcome (creative practice or computing artefact).

Consistent with the appropriated course codes, we asked each student to create an individual laptop ensemble work to be performed at the end of the semester. To help fill in knowledge gaps for both groups of students we provided a crash-course in computer music through a four week seminar series at the beginning of the semester covering dig-

ital synthesis, algorithmic composition, interfaces, interaction and live coding. The first three topics were taught using Pure Data (https://puredata.info), while Extempore (https://github.com/digego/extempore) was used for the live coding material. We added bi-weekly process diary entries, a graded design motivation document, and performance at a work-in-progress concert (See Figure 3) to allow more opportunities for feedback to the students on how they are tracking.

This iteration of LENS has been broadly successful, with each student developing a composition and performing in the work-in-progress and final concerts. In contrast to version 1, we are able to assess the students creative and technical output, and to give feedback along the way. With four CS students and four music (composition) students involved, the LENS contains a mix of experience with computing and creative skills with both groups challenged to fill in their knowledge. The cracks in this iteration are related to the volume of material to be covered in order to create, implement, and perform a laptop ensemble piece. The crash-course could easily be a full course (as it is in many other institutions) and another full course could be devoted to the collaboration, performance and staging issues of computer music.

6. CONCLUSIONS AND QUESTIONS

Our attempts to bring musicians and computer scientists into a laptop ensemble have challenged us as educators, as well as challenging students on both sides. Similarly to other laptop ensembles we have experienced the tension between content delivery, workshop or studio time, and actual rehearsal of the performative outcomes. We see benefits for both cohorts of students, but it could be that computing students, starved of credit-earning creative courses, see the greatest benefit. Our attempts have brought us on a tour through multiple course codes; and while the present iteration is likely to have the most completed student works, we are already imagining what a more inclusive LENS course could look like.

The laptop ensemble as a phenomenon seems unlikely to go away. Music technology is now (probably, finally) unavoidable as a relevant focus of study in the academy. Computer science and "coding" has surged ahead as part of a new anxiety regarding STEM in all levels of the education system but concerns over how to shape well-rounded computer scientists and engineers may lead towards more focus on creative arts within these fields. While we acknowledge the danger of an creating an electroacoustic aesthetic wasteland, we feel that developing collaborative communities of practice with multi-disciplinary students is worth the risk, and the challenges of navigating university bureaucracies.

While we feel ill-equipped to conclusively evaluate our efforts, we present a number of open questions that have arisen so far in our work and that we are using to guide our ambitions for future iterations of LENS.

- How do we balance allowing the CS & music students to play to their strengths (tech, music respectively) while also encouraging (forcing) them to engage with the "other"[1] side of the course?
- How do we fairly assess group work in a heterogeneous student group, with very different experiences & expectations with regard to assessment, notions of quality/correctness/*goodness*?
- Can we scale the LENS experience to meet demands of large student numbers; in other words, can we run a LENS class with 50 students? What about 500?
- Can a LENS class give rise to a LENS *scene*? Can a course for credit interact successfully with an emergent artistic culture?

7. REFERENCES

R. Beiter, R. Nash, M. McCrady, D. Rhoades, M. Linscomb, M. Clarahan, and S. Sammut. The prevalence and correlates of depression, anxiety, and stress in a sample of college students. *Journal of Affective Disorders*, 173:90–96, Mar. 2015. doi: 10.1016/j.jad.2014.10.054.

J. Bischoff, R. Gold, and J. Horton. Music for an interactive network of microcomputers. *Computer Music Journal*, 2(3):24–29, 1978. doi: 10.2307/3679453.

I. Bukvic, T. Martin, E. Standley, and M. Matthews. Introducing L2Ork: Linux laptop orchestra. In *Proceedings of the International Conference on New Intstruments for Musical Expression*, pages 170–173, 2010.

L. Cheng. Musical competency development in a laptop ensemble. *Research Studies in Music Education*, 41(1):117–131, Apr. 2019. doi: 10.1177/1321103X18773804.

A. Harris and L. de Bruin. Steam education: Fostering creativity in and beyond secondary schools. *Australian Art Education*, 38(1):54, 2017.

A. Hunter, Y. Z., B. Harb, M. Watson, and T. Morrill. Sounding canberra: One, 2018. URL https://collectedresonances.bandcamp.com/album/sounding-canberra-one.

S. Knotts and N. Collins. The politics of laptop ensembles: A survey of 160 laptop ensembles and their organisational structures. In *Proceedings of the International Conference on New Intstruments for Musical Expression*, pages 191–194, 2014.

C. P. Martin. Percussionist-centred design for touchscreen digital musical instruments. *Contemporary Music Review*, 36(1-2):64–85, 2017. doi: 10.1080/07494467.2017.1370794.

M. Rosli, K. Yerkes, M. Wright, T. Wood, H. Wolfe, C. Roberts, A. Haron, and F. Estrada. Ensemble feedback instruments. In E. Berdahl and J. Allison, editors, *Proceedings of the International Conference on New Interfaces for Musical Expression*, NIME '15, pages

[1] although we reject the notion of computer music splitting neatly into a "music side" and a "computers/tech side"

144–149, Baton Rouge, LA, USA, 2015. Louisiana State University.

B. Swift. Chasing a feeling: Experience in computer supported jamming. In S. Holland, K. Wilkie, P. Mulholland, and A. Seago, editors, *Music and Human-Computer Interaction*, Springer Series on Cultural Computing, pages 85–99. Springer, London, UK, 2013. doi: 10.1007/978-1-4471-2990-5_5.

D. Trueman. Why a laptop orchestra? *Organised Sound*, 12(2):171–179, Aug. 2007. doi: 10.1017/S135577180700180X.

D. Trueman, P. R. Cook, S. Smallwood, and G. Wang. PLOrk: The Princeton laptop orchestra, year 1. In *Proceedings of the International Computer Music Conference*, 2006.

P. Valiquet. A managed risk: Mediated musicianships in a networked laptop orchestra. *Contemporary Music Review*, 37(5-6):646–665, 2018. doi: 10.1080/07494467.2017.1402458.

G. Wang, D. Trueman, S. Smallwood, and P. R. Cook. The laptop orchestra as classroom. *Computer Music Journal*, 32(1):26–37, Feb. 2008. doi: 10.1162/comj.2008.32.1.26.

G. Wang, N. J. Bryan, J. Oh, and R. Hamilton. Stanford laptop orchestra (slork). In *Proceedings of the International Computer Music Conference*, 2009.