Current Iteration of a Course on Physical Interaction Design for Music

Sasha Leitman
Center for Computer Research in
Music and Acoustics
Stanford University
1932 Wallamaloo Lane
sleitman@ccrma.stanford.edu

ABSTRACT
This paper is an overview of the current state of a course on New Interfaces for Musical Expression taught at Stanford University. It gives an overview of the various technologies and methodologies used to teach the interdisciplinary work of new musical interfaces.

Author Keywords
NIME, Pedagogy, Design, Education

ACM Classification
H.5.2 [Information Interfaces and Presentation] Input devices and strategies, H.5.2 [Information Interfaces and Presentation] Prototyping, H.5.5 [Information Interfaces and Presentation] Sound and Music Computing, K.3.2 [Computing Milieux] Curriculum

1. INTRODUCTION
An increasing number of institutions are offering courses and entire programs devoted to the design of Digital Musical Instruments. Music 250A at Stanford University's Center for Computer Research in Music and Acoustics is one of the oldest and most continuously running courses devoted to teaching new musical interface design.

NIME Courses are an excellent platform for researching and questioning our musical and technological worlds. As D'Arcangelo notes [3], these courses are a unique opportunity to teach students working at the intersection of hardware, software and music technology. They leverage a wide range of student skills and present students with an opportunity to unite their previous coursework and creative interests.

This paper will discuss the history and current iteration of Stanford's course, examine the value and the challenges of these courses and present a number of teaching strategies that have been found useful.

2. HISTORY, CONTINUITY, RESULTS AND CHANGES
2.1 Course History
Between 1996 - 1999, the course was taught as a collaborative effort between Stanford University, Princeton University and San Jose State University. During those early years, the teaching staff consisted primarily of Perry Cook, Ben Knapp, Bill Verplank, and Max Mathews. Other instructors included Sile O'Modhrain, Dick Duda, David Jaffe, Matt Gorbet, Bob Adams. Funding for the early years of the course came from a National Science Foundation Grant.

At the first NIME conference in 2001, Verplank et al [13], described the course structure and pedagogical results.

Bill Verplank and Max Mathews taught the course from 2001 - 2003. From 2004 to 2006, Bill Verplank was the only official instructor of record but Mathew's continued to be a regular presence in the course. In 2007, Michael Gurevich took over as the instructor of the course for one year before passing it on to Wendy Ju and Edgar Berdahl who taught the course from 2008 - 2012. Sasha Leitman has taught the course from 2013 - 2016 and will hand off the course to Ge Wang for the 2017-18 academic year.

In 2001, a two-week workshop on the subject of Digital Musical Instruments was added to the CCRMA summer workshop curriculum which is open to non-Stanford students. The workshop has been reduced to one week but continues to be offered.

2.2 Course Continuity
Each instructor has added their own research and creative interests to the course and many of the instructors have adapted the tool chain used in the course. However, despite the change of instructors and the long lifetime of the course, the overall arc of the course and many of its methodologies have remained the same.

250A has always been a single-quarter, 10-week, project-based course where the first portion of the quarter is devoted to teaching a technological tool chain and the second quarter is devoted to final project work. The course enrollment has remained relatively static with an average of seventeen students per quarter (Figure 1).

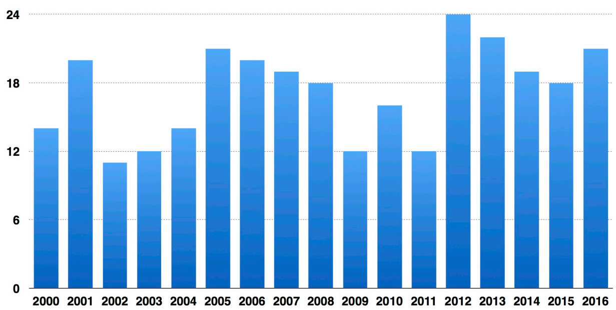

Figure 1: Course Enrollment

Bill Verplank's Interaction Design [12] methodologies have been taught in the course for the last fifteen years and have served as one of the frameworks for ideation and conception in final projects.

NIME'17, May 15-19, 2017, Aalborg University Copenhagen, Denmark.

The Max Lab prototyping lab has been a resource for student projects. The size and equipment of the lab has expanded but the fundamental usage of the lab has remained the same - students have twenty-four hour access to the facilities, are allowed to use parts and materials from the lab for their own projects, and the lab remains a communal space shared with the entire Center.

2.3 Academic Results
The final projects in the course have resulted in papers published at conferences such as NIME, Guthman Music Competition finalists and have spawned work that became integrated into PhD thesis. [1]

2.4 Course Changes
2.4.1 Computer Science Co-listing
For much of the history of the course, the class was co-listed as a computer science course and counted towards a focus in Human Computer Interaction within the computer science program. The current instructor has let this lapse but is hopeful that it can be reinstated.

2.4.2 Tool Chain
The tool chain has changed over time but has generally remained within the same paradigm. Sensors are connected to a microcontroller, which sends serial data to a graphical programming language such as Max/MSP or Pd [1][15].

3. CURRENT ITERATION
3.1 Course Structure
In the current iteration of the course, there is one teacher and a teaching assistant. For the last four years, we have had an average of 20 students and auditors in the course and an average of 11 final projects. The students are a mixture of undergraduate and graduate students studying Music, Art, Electrical Engineering, Product Design, Neuroscience, Symbolic Systems, Computer Science and Mechanical Engineering.

There are two, two-hour blocks of class time and each student is required to come to one, two-hour lab session during the week. The lab sessions are spread out to accommodate a range of schedules. In the first four weeks, lab assignments are performed during the lab times but later in the quarter, the lab sessions function as required attendance at office hours.

The course is 10 weeks long with final projects presented in the last week of class. The first two weeks are focused on the basic tool chain. In the second lab, they create a simple instrument using discrete and continuous sensors. After the second lab, the students have been introduced to each element of the technical tool chain that they will be using. In weeks three to five, lecture topics include greater detail regarding the technical tool chain and an overview of sound design. In addition to these course topics, students brainstorm final project ideas, research those ideas and form final project groups. Week six is devoted to learning strategies for project management, robust building techniques, and a lecture on the physical properties of materials and techniques for building their design. Weeks seven to nine are primarily focused on final project work. During this last period, we present short lectures on additional topics that might be useful in their final project or future designs - alternative outputs such as LEDs, video, and solenoids/motors, Max for Live, different tool chain options such as game controllers or image capture (Figure 2).

Week	1	Overview, Arduino, Electronics
Week	2	Max/MSP, Sensors, Gestures, Serial Communication
Week	3	More Sensors, Start Brainstorming
Week	4	Sound Desing
Week	5	Form Groups, Start Prototyping
Week	6	Robustness, Materials and Project management
Week	7	Guest Lecturer, Alternative outputs
Week	8	Guest Lecture, Max for Live
Week	9	Work on Projects
Week	10	Final Project Presentations, Documentation, Clean up

Figure 2: Course Schedule

3.2 Invited Guests
In addition to the course schedule presented above, each year two guest speakers are invited to lecture on their own work with DMIs. Previous guest lecturers have included Roger Linn, Ean Golden, Matt Moldover, Victor Gama, Kiran Ghandi, and John Aquaviva.

The guest lecturers offer an opportunity to hear perspectives that differ from that of the teaching staff. The timing of the lectures in the course schedule depends greatly on the availability of the guests but an attempt is made to have them present later in the quarter when they can give feedback to students about their final projects.

3.3 Tool chain
There are many tools, methods and technological approaches available to create a digital music interaction. This course currently uses sensors, the Arduino microcontroller, and Max/MSP. Students write their own serial "protocols" in the Arduino IDE and Max/MSP.

These are tools with large online communities and excellent tutorials - students can find information about problems and applications not covered in the course. They are tools that can be used for work in other contexts - an art student might never use sound again but could use Arduino or other microcontrollers for controlling lighting or kinetic elements. The tools are well suited to mid-resolution prototyping and are commonly used in professional contexts during the design process - while it is not common to see these tools employed in industry products that ship to end-users, they are commonly used in the design of those products.

For the final projects, students are allowed to use any technology that they feel is appropriate.

3.4 Everyone Learns Everything
Each student is expected to learn each part of the tool chain but they are not expected to excel at each portion. Requiring students to learn and implement each aspect of the tool chain ensures that they have a thorough understanding of the course material. Artists learn the engineering aspects of the course and engineers learn the sound design aspects. In final group projects, it is common to divide the labor according to each person's strength. This is an effective strategy but by having a familiarity with each portion of the tool chain, students can more easily communicate with one another.

3.5 Final Project
The course culminates in a final project that is presented during the last class period. Members of the CCRMA community and

outside guests are invited to attend these presentations and we typically have around seventy attendees. The final presentations are a mixture of performances and demonstrations.

4. VALUE OF THE COURSE

4.1 Navigation of Simultaneous Design Demands

Courses on DMIs present a unique opportunity to simultaneously engage the hardware, software and sound design aspects of computer music. Lehrman [9] points out the value of students from different academic backgrounds working together and teaching one another.

In addition to this collaborative cross-pollination, there is another pedagogical benefit to the interdisciplinary demands of these classes. Hardware, software and sound design in DMIs are interdependent and creating a successful instrument demands that the three topics are simultaneously addressed and considered. The need to balance the three demands and remain cognizant of how changes in one aspect will influence the other two aspects is a cognitive challenge that requires a thorough understanding of each element (Figure 3).

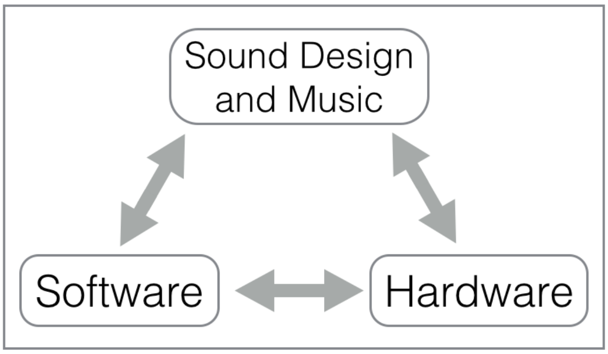

Figure 3: Simultaneous Design Demands

4.2 Breadth and Cohesion

The merging of an artistic/musical endeavor with so many technical concerns is an opportunity for students to both broaden and deepen their knowledge. The students who take the course can be grouped into roughly three categories: 1) Undergraduate and graduate students who are studying for degrees at CCRMA in Music Science and Technology (Undergraduate and Masters) or Computer-Based Music Theory and Acoustics (PhD) 2) Engineering students who wish to apply their knowledge to musical applications and 3) Art and Music students who are looking to gain technical skills to apply to their creative work.

For the students who are deeply engaged with computers and music, the course offers an opportunity to fully engage their knowledge. By working with hardware, software and sound design, they have an opportunity to synthesize their learning from other courses. Advanced students are able to explore complex ideas and projects that leverage their knowledge and experience to create designs that incorporate their specific interests.

For undergraduate and graduate students who are working towards degrees focused on Computer Science or Engineering, the course provides a platform to apply their knowledge to musical applications. Music provides a unique set of design constraints and a new avenue for applying their engineering skills.

For Music and Art students, the course is an excellent introduction to both music technology and tools for creating interactive, sound and installation art. An introduction to sensors, microcontrollers, electronics and basic sound design can be applied to any range of projects that they might want to pursue in the future.

4.3 Real World Design Application

Creating a working interaction device strengthens students problem solving and design skills. While there are many prototyping exercises and benchmarks throughout the course, the end goal for final projects is a working first draft of their creation. Creating a fully functioning device and considering the hardware, software and sound design demands of this device, requires fully engaging real world materials and constraints. For example, it is one thing to learn about the basics of electronics and sensors and another to experience their limitations and nuances. In the case of sensors, this might mean accounting for nonlinearities and error cases, by adapting the physical object that is being sensed to prevent unwanted signals, applying smoothing techniques to the software, or accommodating the sound design to take advantage of those unexpected signals. Each of these reactions to the unwanted sensor results has a different effect on the outcome of the project - it is up to the students to find a solution that corresponds to their fundamental design goals. Addressing these unexpected challenges, cements students knowledge of the technology and strengthens their ability to make choices regarding technology, aesthetics and interactivity.

4.4 Portfolio Project

In addition to the pedagogical benefit of the course, the final projects often become an important part of a student's portfolio. The work demonstrates the ability mentioned above to engage hardware, software and sound design. Additionally, it provides an opportunity for students to demonstrate the aesthetic and conceptual issues that interest them. In creating a fully functional device, they are able to demonstrate the breadth of their knowledge and abilities.

5. CHALLENGE OF THESE COURSES

5.1 Interdisciplinary, Interdependent Subject Matter

A wide range of topics must be addressed in a course on DMIs. While these disciplines do not need to be learned in great depth, a basic understanding of the following topics is required for the successful creation of most DMIs:

Electronics (electronics and sensors)
Computer Coding (Coding for Arduino and Max/MSP)
Sound Design (Synthesis, Audio Effects and Audio Signal Flow)
Fabrication (Materials and strategies for robust physical design)
Product Design (Designing instruments that engage significant questions or concerns of interaction design within the context of musical instruments.

5.2 Diverse Student Backgrounds

Students in this course come from diverse backgrounds and have a wide range of goals in taking the course. Technical skills vary widely from students with little to no knowledge of computer programming or electronics to graduate students in engineering. There is an equally large gap in student experience discussing conceptual, aesthetic and design criteria.

If harnessed correctly, this diversity is a strength in that it allows students to learn from one another. The primary pedagogical challenge is finding a way to teach the various course topics in a way that engages the beginners without causing the more experienced students to lose interest. Various strategies will be discussed later.

5.3 Conceptual and Design Questions

A fundamental question arises in this course - it is it enough to teach students a technical and musical tool chain or do questions of design, aesthetics and concept also need to be pursued?

In the heat of creation, practical considerations can overwhelm any creator. It is a challenge to keep students asking meaningful questions about their designs and pursuing new avenues of inquiry. The DMIs created in the course operate within a broad historical context and a contemporary context which is constantly evolving. The curriculum must communicate some of that context and also give students tools and prompts that allow them to explore the context specific to their designs. Various strategies for engaging these questions will be discussed later.

5.4 No Textbook

A further challenge is the lack of an established text for this course material. There are books that look at portions of the technology but none that address the issue directly through the lense of musical interaction [5][11]. Readings that address an overview of the field are assigned from a number of sources including several frequently cited NIME articles [2][14]. These are excellent articles but they are directed at experienced practitioners, not students new to the field.

6. CURRENT TEACHING STRATEGIES

A number of pedagogical strategies help to address some of these challenges.

6.1 Quick Startup

Concepts and techniques discussed in class are implemented during lab assignments. By week two, the lab involves essentially all of the major components for a working instrument. Students connect a discrete and a continuous sensor to the Arduino microcontroller, write Arduino code to send the sensor data via serial over USB, and create a Max/MSP patch to parse the data and create sound. This requires technical details such as breadboarding, voltage dividers, serial protocols and two different kinds of sound interaction.

Accomplishing the bare minimum technological proficiency so early in the course gives an overview of the necessary technological components, demystifies the primary elements of the technology, allows the rest of the quarter to be spent refining and deepening knowledge of each part of the process, and gives an example of a possible technical structure for final project ideas.

6.2 Self-guided Learning – Electronics

One of the most striking examples of the disparity of knowledge discussed above is in the area of electronics knowledge. In previous iterations of this course and in others that I have taught, we would give a 1 - 2 hour lecture on electronics. This seemed to be the worst of both worlds - students who were familiar with the material would “tune out” and ignore certain small details that while they were not emphasized in traditional electronics courses are very important for work with sensors while students who were new to the subject were overwhelmed with new information. As a teacher, I wavered between rushing the lecture to prevent boredom for advanced students and over emphasizing certain details to the point of boring even the students new to the material.

Clearly, a one to two hour lecture is not enough time for students new to electronics but it is far too much time for students who have had significant experience. In the last two years, I have developed a useful remedy.

In the first week, I give students a copy of an Electronics quiz with the answers printed below the questions. The questions cover the primary electronics concepts that are necessary to understand for the technology used in the course - Voltage, Current, Resistance, Ohms Law, Series vs Parallel circuits, Multimeters, Pull-up Resistors, Voltage Dividers, and Schematic symbols.

Under each question, there are links to online tutorials that cover the topics in depth. If a student does not know an answer, they can go to the tutorials and spend as much time with the material as they personally need.

The quiz (without the answers) is given in the second week. The teaching staff work with any students who have trouble with the quiz to make sure that they learn the material.

The result of this teaching method has been that students have a solid grasp of the knowledge necessary for their projects. In the two years that we have used this method, there has been a dramatic reduction in confusion about basic electronic principles during the final projects.

6.3 Final Project Strategies

A number of pedagogical choices have been made in the final project prompt.

6.3.1 Any Technology To Achieve the Goal

Students are allowed to use any technology they choose for their final project. In the interest of time, we can only cover one primary tool chain in the course material. Different options are mentioned briefly in lectures but not covered in great depth. By opening up the final project to different technologies, we allow students to leverage their previous knowledge while making clear that the technology should not be the primary focus of projects - it should be a tool to create the intended interaction and there are many tools available to designers of DMIs.

6.3.2 Continuous Sensors

There are many examples of commercial music controllers that consist of many discrete buttons. While these devices are excellent for making computer music, there are already many of them available and their use cases are fairly well established.

In order to guide students towards asking more nuanced questions in their projects and to steer them away from designing instruments that are fundamentally sample playback devices, we require that the primary component of interaction is a continuous gesture.

6.3.3 *Questions to Consider*

Students are asked to design a final project around a question they would like to explore or a problem they would like to address. Framing their ideas around these structures brings them back towards conceptual thinking about their designs. As mentioned above, the demands of a complex project can distract students from aesthetic, conceptual and design inquiries.

In addition to this question, each group must answer other questions about their design such as: Who can use it? How long will it take to learn? Can the user develop a model of the sound space and achieve a desired sound? Is the mapping clear for the performer and/or audience?

6.3.4 *More is More - Lots of Ideas, Lots of Research*

The first development assignment for final projects happens in week three - students are required to sketch ten ideas for their final project. In week four, those ideas are presented to the class. It is a dizzying experience to hear two-hundred ideas presented in a hundred and ten minutes. This assignment exercises students' creative abilities and makes it less likely that they will become attached to the first idea they have. Often it is the third or seventh sketch that becomes their final project. Additionally, the presentation of these ten ideas fosters the creation of groups. Students are able to find collaborators that share similar threads of interest.

The second development assignment for the final project is to take their three favorite ideas and find ten examples of similar work; it can be similar in concept, technology or aesthetic. It can be comparable to the project idea as a whole or to one component of the project idea. This assignment forces students to clearly identify the salient parts of their ideas and builds their understanding of their idea within the context of prior work.

6.3.5 *Group and Solo work are acceptable*

The project can be done solo or in groups. As a generalization, groups tend to work better for creating tools that other people will use. Working alone seems to work best for people who have a specific creative project or composition for which they would like to some physical interaction component. A large number of individual projects makes many things more difficult for the teaching staff. However, we feel that giving students the opportunity to create a DMI specific to their personal creative practice is worth the extra labor.

6.4 Context and Example

In order to establish a context for the work of creating DMIs, the course relies on a mixture of readings, examples and resources. Each week, an article or book chapter is assigned and discussed in class. The readings might present an overview of strategies for designing interfaces[2][14] or it might explore a specific topic such as sonification. Most classes feature examples of prior work and an effort is made to find examples relevant to work being done by individual students or groups. The course website features links to websites where more examples can be found such as the NIME proceedings page, Create Digital Music, Instructables, etc.

When students present their project ideas, the instructor responds individually or in a group email with examples that relate to the specific ideas.

6.5 Prototyping and Design

6.5.1 *Sketching*

Sketching is an important skill in design and the course begins with a sketching exercise on the first day of class. As a way of introducing themselves to the group, students are asked to draw a picture of their favorite instrument, an instrument they would be interested in and able to create within the context of the course, and an instrument that they would make if money, time, skills and even the laws of physics were not a barrier. It is not unusual that the idea they think they could make is actually prohibitively difficult, and the idea that they thought would be too hard is relatively simple. This first sketching exercise is an opportunity to guide them in sketching techniques such as including elements in the sketch that display interaction such as a hand, foot or lips.

In addition to sketching their ideas, the students are asked to act out the use of their instrument and vocalize the sounds that they envision the device making. In addition to being entertaining and a good "ice breaker", this activity sets the expectation and begins the practice of thinking about both the physical and sonic interaction from the start of the ideation process.

Sketching assignments in class and in homeworks continue throughout the duration of the course.

6.5.2 *Design Prompts*

In addition to sketching, other prototyping processes are employed in the final project. A low-resolution cardboard prototype is made in class to estimate ergonomics and gestures. A storyboard graphic chart is made that aligns events, gestures and the sounds that accompany those events and gestures.

A design criteria list is created - the entire class brainstorms the things that they like about particular musical instruments. The list typically fills two large whiteboards and when it is finished, a spreadsheet of that list is shared with the class. One column of the spreadsheet is the list of criteria and each subsequent column has the name of a student. Each student must mark the criteria that they value in general and the criteria that they hope to achieve in their particular final project. This list becomes a touchstone during future critiques and evaluations.

6.5.3 *Design Models*

A number of design methodologies are discussed in class with a particular emphasis on Bill Verplanks Interaction Design framework being most prominent. These methodologies and models guide students through different thought processes and strategies. They provide a framework for examining and developing their ideas.

6.5.4 *Project Management*

An effort is made to help students develop their project management skills. The two most important strategies are 1) Identifying the minimum viable product - the simplest iteration of their idea that would still accomplish their design goals - and the features that would be nice to have but not necessary. 2) Creating a detailed to do list for the project with clearly dated benchmarks and a list of materials or supplies necessary for the accomplishment of those benchmarks.

Clear articulation of these two items helps the students accomplish their goals but it also supports the teaching staff's ability to help the students' progress.

6.6 Mandatory Office Hours (aka Lab Times)

In the first four weeks, lab assignments are performed during the lab sessions but later in the quarter, lab sessions function as required attendance at office hours. This is an important tool for keeping students on track with their final projects and making sure that they are not wasting time on technical difficulties that can be quickly solved by the teaching staff. Students are often tackling problems that they have never faced before - from using a drill to computer coding.

6.7 Documentation

Documentation of projects is mandatory and begins at the start of ideation. The course website uses wordpress which allows students to create pages. Each student creates a page for their 10 idea sketches, that site becomes the place where they post research for three of those ten ideas. When final projects and groups are created, students create a new wordpress site for the project. Each week, there are documentation requirements for the final project - take ten pictures, add five sketches, post your to do list, etc. Additionally, the teaching staff can leave notes on the site pointing students towards relevant materials or examples. The site is not intended to be viewed by outside guests but is rather a place to store materials for later documentation or reference. It traces the story of the projects design and creation.

After the final project presentations, students sign up for documentation, wrap up and clean up sessions. The projects are filmed in a well-lit and quiet environment and the teaching staff edits the videos to create a video for each project. This is also a time to discuss the project outcomes with the students and answer any questions about future development. Finally, the documentation sessions are a chance to give each student a cleaning task in the shared prototyping space. These cleaning tasks prepare the lab for the next class and make sure that everything is stocked and ready to go.

7. Course Goals, Evaluation, and Assessment

On one hand, assessing these courses is very difficult – an evaluation of learning goals is dependent on the skills that students bring to the course and that skill set is extremely diverse for such an interdisciplinary course. On the other hand, the portfolio quality and completion rate of the projects can be seen as concrete evidence of student growth since few students enter the course with a complete skillset necessary for the course. While the sophistication and complexity of the projects vary, in the last four years, each student or student group has completed a functioning final project.

Evaluation of student work is done using a model common to courses in both Design and Art using a combination of both group and individual critiques. All critique sessions reference the students' creative and design goals as articulated by brainstorming and project planning exercises such as the list of desired attributes mentioned previously. This allows feedback and assessment to be tailored to students' abilities, interests and backgrounds. It also engages pedagogical concepts such as metacognition and strengthens students' ability to evaluate and refine their own work.

The course receives positive reviews both in the official course evaluations and in later informal interactions with students. More concrete research could be done into skill acquisition, intellectual development and specific desired benchmarks. Studies such as these would provide extra justification for NIME courses.

8. CONCLUSION

It is exciting to see the number of courses on NIMES, DMIs, and Physical Interaction Design growing. This paper has highlighted a number of approaches that have been useful in one academic context. Each practitioner of this field will have a different approach and each teaching environment will pose different challenges.

These are challenging courses to teach but they are an excellent opportunity for students to pursue questions of hardware software and music within the context of interaction.

9. REFERENCES

[1] E. Berdahl and W. Ju. Satellite CCRMA: A Musical Interaction and Sound Synthesis Platform. In Proceedings of the 2011 conference on New interfaces for musical expression. Oslo, Norway, 2011.

[2] P. R. Cook, "Re-designing principles for computer music controllers : a case study of squeezevox maggie," in Proceedings of the international conference on new interfaces for musical expression, Pittsburgh, PA, United States, 2009, pp. 218-221.

[3] G. D'Arcangelo, "Creating a context for musical innovation: a nime curriculum," in Proceedings of the international conference on new interfaces for musical expression, Dublin, Ireland, 24-26 May, 2002 2002, pp. 46-49.

[4] "M. Gurevich, B. Knapp, and S. Jorda`. A Workshop on NIME Education. In Proc. of NIME, 2011."

[5] T. Igoe. Making Things Talk: Using Sensors, Networks, and Arduino to see, hear, and feel your world. San Francisco, CA: Maker Media, Inc. 2011.

[6] S. Jordà and S. Mealla, "A methodological framework for teaching, evaluating and informing nime design with a focus on mapping and expressiveness," in Proceedings of the international conference on new interfaces for musical expression, London, United Kingdom, 2014, pp. 233-238.

[7] Jensenius, R. J. "An action-sound approach to teaching interactive music" Organised Sound. 18/2 August 2013, pp. 178-189.

[8] A. Kapur and M. Darling, "A pedagogical paradigm for musical robotics," in Proceedings of the international conference on new interfaces for musical expression, Sydney, Australia, 2010, pp. 162-165.

[9] P. D. Lehrman and T. M. Ryan, "Bridging the gap between art and science education through teaching electronic musical instrument design," in Proceedings of the international conference on new interfaces for musical expression, Vancouver, BC, Canada, 2005, pp. 136-139.

[10] K. Moriwaki and J. Brucken-Cohen, "Midi scrapyard challenge workshops," in Proceedings of the international conference on new interfaces for musical expression, New York City, NY, United States, 2007, pp. 168-172.

[11] D. O'Sullivan & T. Igoe. Physical Computing: Sensing and Controlling the Physical World with Computers. Tompson, 2004.

[12] B. Verplank. Interaction Design Sketchbook. http://www.billverplank.com/IxDSketchBook.pdf

[13] B. Verplank, C. Sapp, and M. Mathews. A Course on Controllers. In Proceedings of the CHI'01 Workshop on New Interfaces for Musical Expression (NIME-01), Seattle, USA, 2001

[14] D. Wessel and Wright.. Problems and Prospects for Intimate Musical Control of Computers. Computer Music Journal 26, no. 3: 11-22, 2002.

[15] S. Wilson, M. Gurevich, B. Verplank and P. Stang. Microcontrollers in Music HCI Instruction: Reflections on our Switch to the Atmel AVR Platform. In Proceedings of the 2003 conference on New interfaces for musical expression. Montreal, Canada, 2003.