ABSTRACT
In this paper, we describe an ongoing multidisciplinary undergraduate seminar that we have developed, in which student teams build non-human systems that conduct our college orchestra. Students and faculty in the course come from four disciplines: computer science, interactive multimedia, music, and mechanical engineering. This paper describes the course structure, computer science components, final projects, team dynamics, and assessments. We evaluate the results to-date and discuss ongoing revisions and expectations for the future.

Categories and Subject Descriptors
K.3.2. [Computers and Education]: Computer and Information Science Education – Computer education; J.2. [Computer Applications]: Physical Sciences and Engineering – Engineering; J.5. [Computer Applications]: Arts and Humanities - Music

General Terms
Algorithms, Design, Human Factor.

Keywords
multidisciplinary, robotics, interfaces, music, orchestral conducting

1. INTRODUCTION
As we advance into the 21st century, computer science is increasingly playing a central role in other disciplines, from science and engineering to humanities and arts. Nearly all areas of academic and scholarly endeavor now rely on computer software for presenting and conducting academic research. Future generations of scientists, engineers, and artists will benefit from understanding and using principles of computer science and computational thinking [12, 3].

Many of the most pressing problems of the day are best solved using interdisciplinary approaches [8]. However, the standard paradigm in interdisciplinary work is that it requires participants to become fluent in all of the participating disciplines, sometimes sacrificing depth for the sake of breadth. A multidisciplinary approach modifies this scenario through the participation of specialists from different fields who have the necessary skills to communicate with each other and produce a collaborative solution through a common perspective [2].

Regardless of their subject discipline, 21st-century undergraduates will be better served by a multidisciplinary approach to education, to prepare them for future tasks and environments. Non-computer science majors should become familiar with concepts of computational thinking, while computer science majors should learn to communicate with specialists from other disciplines. Recently, a wide variety of interdisciplinary approaches to computer science education have been developed and described in the literature, ranging from entire programs [6] to interdisciplinary courses that combine computer science with computer games [13], robotics [11], science [4, 2], media [6], and music [5].

In this paper we describe our own implementation of the multidisciplinary approach through an undergraduate seminar entitled "Conducting Robots." Our course brings together students majoring in computer science, mechanical engineering, interactive multimedia and music in a common setting. It is a project-oriented course that fosters critical thinking, creative problem solving, and computational thinking skills through an open-ended team project requiring the synthesis of knowledge in all four core disciplines. Students work collaboratively to design and develop innovative robotic and graphical conducting systems that can direct an orchestra. Topics taught include robotics, visual music, abstract animation, computer vision, algorithms, data processing, music conducting, and project management. We discuss the course structure in Section 2, and the computer science component in Section 3. We describe some of the students’ final projects in section 4, team dynamics in Section 5 and conclude in Section 6.
2. COURSE STRUCTURE

The Conducting Robots course is an ongoing one-semester course that has been offered twice, in the fall and spring semester of the 2009-2010 academic year. Both times the course was cross-listed in the departments of Computer Science, Interactive Multimedia, Mechanical Engineering, and Music, and was taught by a team of four instructors, one from each department (the authors). Throughout both semesters we worked with an independent evaluator to develop and administer student surveys and interviews. In addition, students were asked to keep a reflection journal. From the first weeks of the first semester these tools were used to evaluate the course and modify it based on student feedback.

The main challenge was designing lectures and assignments for students with four different disciplines as a background. All the students were fairly advanced in their own major (junior or senior level), and had varying amounts of knowledge in the other disciplines involved in the course.

Initially, the course was planned as a two-semester course, with a design phase in the fall semester, followed by an implementation phase in the spring semester. The fall semester was going to be filled with lectures in the four subject areas that would provide necessary background in all the four disciplines. The end of the semester was to be spent on the design of an initial prototype that was going to be implemented and refined in the spring semester.

However, early in the fall semester it became apparent that the majority of students would not be able to register for two semesters due to requirements of their majors. Moreover, they were eager to start building prototypes and get hands-on experience. Traditional lectures and assignments in specific subject areas (especially computer science and mechanical engineering) were either difficult for students majoring in other areas, or boring for students already familiar with the field.

Therefore, in the middle of the semester we decided to change the structure of the class and allow students to work in interdisciplinary teams to develop a non-human conductor. Five teams were formed, and with the exception of the mechanical engineering majors (of whom only three were enrolled in the course), each major was represented on each team.

The second time we offered the course (in spring 2010), the number of traditional discipline specific lectures and assignments was significantly reduced. Their content was changed to be less dependent on in-major background, and more relevant to the course-specific project. Students formed (larger) interdisciplinary teams from the first week and started brainstorming design ideas right away. Whether they were working on the final project or on a smaller assignment, they were expected to rely more on their teammates for problem solving, and therefore become more involved in tutoring each other.

Since many of the students were not very familiar with classical music, numerous opportunities were provided to give them exposure. In addition to the regular weekly class meetings, students were required to attend one rehearsal session of the TCNJ Orchestra. The instructors arranged for a group trip to see the Philadelphia Orchestra conducted by Charles Dutoit at the Kimmel Center in Philadelphia. Additionally, the conductor of the TCNJ Orchestra served as a consultant on this project, and made himself personally available to the students for interviews and discussions about conducting technique.

We also invited guest speakers that provided expertise in different areas required by our project.

3. COMPUTER SCIENCE COMPONENT

In the first semester we introduced the students to computer science and programming through Matlab. The reason for this was that Matlab was required for some of the mechanical engineering content, including forward and inverse kinematics. Students started with simple matrix operations and plotting, and continued with control structures and functions. Due to the limited amount of time, this material was covered in two 3-hour combined lecture/lab sessions. Students worked on two individual assignments that were not related to the conducting robots topic.

Based on student feedback and performance, our approach was different in the second semester. Students continued to use Matlab for mechanical engineering assignments, but this time without control structures and functions. Computer science was introduced through Processing, a Java-based open source programming language and environment for people who want to program images, animation, and interactions [9]. This approach allowed for an easy introduction of algorithms and control structures, as well as computer-based drawing and animation in one 3-hour session. Students worked on a music animation assignment, and while the assignments were individual, they were strongly encouraged to seek help from their computer science major teammates.

Reflection journals and average assignment grades show that the second approach was more successful. Students found it more compelling, as it allowed them to explore computer science in a project-related context. Table 1 shows the average grades obtained by the students on computer science assignments by their major. Not surprisingly, due to their prior background, the best results were obtained by the computer science majors. They were followed by the interactive multimedia majors, due to the closeness of their field to computer science. The weakest results were obtained by the music majors, who were not always able to finish all the requirements, but showed good effort nonetheless.

<table>
<thead>
<tr>
<th>Major</th>
<th>Fall</th>
<th>Spring</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer Science</td>
<td>92%</td>
<td>96%</td>
</tr>
<tr>
<td>Interactive Multimedia</td>
<td>91%</td>
<td>96%</td>
</tr>
<tr>
<td>Mechanical Engineering</td>
<td>74%</td>
<td>90%</td>
</tr>
<tr>
<td>Music</td>
<td>77%</td>
<td>82%</td>
</tr>
</tbody>
</table>

In addition to being introduced to algorithms and control structures, students gained hands-on computer science experience through their final project. Music majors worked together with computer science majors to develop an abstract description of conducting gestures, and of the music that needed to be conducted. Most teams decided to work with the MIDI file format, which they approached through the Finale music composing and notation software. Some teams wrote their own MIDI parsers, while others used existing sample code. Then computer science majors worked together with interactive multimedia majors to create live animations, and with mechanical engineering majors to program the microcontrollers controlling the robots. Programs were written in a variety of languages, including Java, C, and Processing.
4. THE FINAL PROJECT
The final objective of the course was to have students build non-human conductors based on graphics and robots that could conduct an orchestra. The minimum requirement was that each system should indicate beat patterns, tempo, dynamics and cueing to a human ensemble in real-time.

During the Fall, student teams were formed somewhat late in the semester (around week seven). As a result, the final project deliverables were relaxed a little bit. The 20 students enrolled in the class were divided into five teams. The teams without engineering majors focused on graphical conducting systems, while the other teams built robotic conductors complemented with graphical visualizations. The robots were built using Lego Mindstorms NXT kits.

Students were encouraged to build general systems that could conduct any musical piece, but for practical purposes (slower tempo, simplicity of cueing, etc.) the conductor of the TCNJ orchestra suggested Beethoven’s 7th Symphony, Movement 2. The conducting systems were required to conduct at least one minute of the piece.

Five different prototypes were developed. Testing occurred during two special sessions of our college orchestra at the end of the semester. The quality of these prototypes was judged by the performing musicians who gave scores on a scale of 1 to 10 for the systems’ tempo, dynamics, section cueing, conducting style/beat pattern, articulation and level of interaction. Due to the small size of the robotic conductors and the nature of the musical score, only the string section (violins, violas, and celli) of the orchestra attended the rehearsals (25 musicians). The movements of the robotic conductors were also projected on a big screen for visibility. Although it was the first time the course was offered, and it was the musicians’ first experience with a robotic conductor, all of the systems worked and obtained satisfactory scores from the orchestra.

Figure 1 shows three of the five conducting systems developed by students in the first semester. C3 (Cybernetic Conducting Contraption, top) was built using the Lego Mindstorms NXT robotic kit, and had two arms. The right arm had two degrees of freedom and was used to conduct beat and dynamics; the left arm had one degree of freedom and was used for cueing. C3’s functionality was constrained by the limitations of the NXT kit, which can control only up to three motors.

Ahasimo (bottom left) was a non-humanoid graphic conductor implemented in Processing. It conveyed beats by means of two pulsing circles, while dynamics were controlled by the bar on the right. Its cueing functionality was modeled after the Guitar Hero graphical interface.

GUS (bottom right) was a humanoid graphic conductor implemented in Maya. Just like the robotic conductor C3, GUS used its right arm to conduct beat and dynamics, while the left arm was used for cueing. Each arm had three degrees of freedom.

The second time we offered the course (Spring 2010), teams formed in the first week of the semester and started brainstorming design ideas right away. This time the teams were larger, with 17 students divided into three teams, each having two mechanical engineering majors. Each team was required to work on systems with both a mechanical and a media component that had to be complementary, rather than redundant. Robots had to be at least half human size, to be visible by the orchestra. Each team picked its own musical piece to conduct at a public concert event with the college orchestra at the end of the semester.

![Figure 1. Prototypes built by students in the first semester: C3, Ahasimo, and GUS](image)

Just like in the previous semester, the robots were evaluated in a demonstration with the string section of the orchestra, and the musicians gave satisfactory scores to all the robots.

Seven of the 17 students enrolled in the spring class also took the fall class. We allowed them to do so because of our initial plan to have a two-semester course. In the future we will not allow students to take the class for a second time.

Figure 2 shows the prototypes built by the students in the second semester. The students in team C3 from the fall continued in team C4 in the spring and, although they built a new robot (top right), they benefitted from their experience in the Fall semester. The C4 robot was still humanoid (like C3), but was built using the VEX robotic kit and the Arduino microcontroller. Each arm had three degrees of freedom, and they used a mirrored motion to indicate the beat pattern and tempo. Dynamics were indicated by increasing the size of the gesture of the right arm while the left arm would be raised or lowered. Cues were indicated by the left arm pointing at the appropriate section. Articulation was communicated by the robot with both arms, the smoothness of the gestures corresponding to articulation. The robot also featured two real-time procedural animations implemented in Processing: an animated face for the orchestra, and an additional animation on its back to entertain the audience.

This robot was the only one that could “hear” the orchestra, using the Chuck audio programming language for real time audio processing. The motion of the robot and the facial gesture are adjusted according to the tempo and dynamics played by the orchestra.

ACRE (top right) was the robot with the most advanced mechanical component, due to the face that the engineering majors on the team were working on this as part of their senior design project as well. ACRE is a human size humanoid robot built from scratch without the use of any kits, and uses the Arduino microcontroller. It uses its right arm to indicate the beat pattern, and the left arm to indicate cueing and dynamics.
Instead of using the score of the music, the robot learned to conduct from a video of a human conducting the same piece. The hands were tracked in the video using the system described in [7]. Inverse kinematics was used to define a robot motion that would mimic that of the human conductor.

The robot simulates knee bending through the use of a linear actuator in the lower body, and could turn to face different orchestra sections. In addition, the robot’s head displayed a human face that expressed emotional changes according to the score.

The third robot, Superconductor (bottom) was not humanoid. It consisted of a “baton” that rotated on a square board with four LED lights. The rotation of the baton, together with the lights, conveyed the beat pattern. The lights could move closer together or further apart suggesting the dynamics. The team composed its own music to highlight the features of the robot.

Figure 2. Prototypes built by students in the second semester: C4, ACRE, and SuperConductor

5. TEAM DYNAMICS

Students worked in multidisciplinary teams of four in the fall, and six in the spring. Every discipline was represented on each team, with the exception of two teams in the fall that did not have any engineering majors.

Students were asked to keep biweekly confidential reflection journals that documented whether they were aware of “aha” experiences, and who they learned from throughout the semester. The journals were collected by an independent evaluator who also summarized the results for the instructors.

These journals show that teamwork changed students’ perception of their own, as well as the other, disciplines. Music majors started out intimidated by the technical majors to whom they referred to as “smart” majors. The realization that these majors didn’t have any background on music and conducting was an important boost to their confidence. Students saw quickly that conducting systems could not function without musical expertise. While engineering and IMM majors came in knowing exactly what their task would be (build a robot or create a visualization of the music), they had to work with the music majors to understand the functionality of the system that was being built.

While all the instructors understood from the very beginning the major role that computer science would play in the project, students were less clear on this point. Both engineering and IMM students came into the class with some knowledge of programming, and believed they were able to put their systems in motion. The presence of the computer science major seemed unnecessary. In addition, at the beginning of the semester computer science majors did not have the technical background necessary to build a robot or to create a visualization, so some of them started out by feeling that their presence was not required for a successful project.

As the semester progressed, students soon realized that computer science was the link that held all the pieces together. A conducting system uses music to guide its movements. Whether it is the actual musical score (possibly with a human conductor’s annotations) or information about how a human conductor moves his hands to conduct the piece, the language in which this information is represented differs greatly from the language that controls the artificial system. The role of computer science then is a major one: convert the musical representation into one that can be used by a robot or visualization system.

Since students are required to build systems that can conduct any musical piece, this conversion cannot be done by hand. It requires a general algorithm that can break up any musical piece into small pieces, analyze and convert them into another representation, and put them back together. Students have to use abstraction to solve this problem, as well as knowledge from all the areas involved: music, engineering and multimedia. While the computer science major has some expertise in algorithm design and implementation, working with a multidisciplinary team requires him to communicate in a language that is understood by everyone. Through this process, CS majors gained a better understanding of their own field, while the other students were introduced to the basics of computational thinking.

In addition to reflection journals, we asked students to assess their own and their teammates’ teamwork, using a rubric developed based on [10]. This assessment, together with the instructors’ perception of students’ individual teamwork, was used to determine individual final grades in the course. The attributes in the rubric were:

- Process – includes caring about goals, exhibiting leadership skills, helping the group in setting and meeting goals, and exhibiting consistent on-task behavior
- Communication – includes sharing ideas, encouraging other group members to share ideas, listening skills, incorporating comments
- Interpersonal skills and social interaction – includes involving the whole group in problem solving, actively working together with the group, being aware and respecting the views and opinions of others, empathizing with others’ ideas and feelings
- Contributions – includes contributions to decision making, work, and evaluation
- Responsibility sharing – includes active participation, completion of assigned tasks, and ensuring that responsibilities are evenly assigned

Since the evaluations were anonymous, students did not hesitate to pinpoint those who did not contribute enough to their team.

The change in course structure from the fall to the spring semester was visible in the team dynamics as well. Teams formed and started work earlier, and teamwork was emphasized in assignments throughout the semester. While most students rated
their teammates as excellent both times, more students were satisfied with their teammates performance the second time we offered the course (Figure 3).

![Figure 3. Percentage of teamwork attributes evaluated as “Excellent”](image)

Teamwork issues had to be resolved in the instructor team as well. Given the fundamental differences between science, engineering, and the arts, instructional and grading styles are very different. These differences were sometimes perceived by the students (as indicated in their anonymous reflection journals) as unfair. Arts students felt the lectures were tough and boring when engineering/science topics were delivered by engineering/science instructors. These topics were usually introduced in terms of definition/concept/principle and explained through logic derivation and application examples with proper math background. At the same time, engineering students thought they learned little in arts topics as these were usually introduced through students’ discussion in class and often there was no right or wrong answer.

6. CONCLUSIONS AND FUTURE WORK

We presented the first two semesters of teaching an on-going undergraduate multidisciplinary seminar entitled “Conducting Robots”. Through this seminar, majors from four disciplines, computer science, interactive multimedia, music and mechanical engineering, worked together contributing their own expertise to multidisciplinary teams concentrating on a common goal: building a non-human system that could conduct the college orchestra.

Throughout the two semesters we used student feedback collected by our independent evaluator to reflect on and revise the course. We have found that our original plan to teach the seminar over the course of an entire year was unrealistic given the graduation requirements of our students, and we adapted to the shorter schedule using a hands-on approach that was favored by the students.

Designing lectures that are engaging and challenging to students from such varied backgrounds was another obstacle that we encountered. We tried to keep in-class discussions general, and allowed students to discover and teach each other the details.

Due to the multidisciplinary collaboration our students were able to build working systems at a level that is rare in an undergraduate setting, and is more characteristic of graduate research.

We plan to offer the course two more times, after which we believe that the novelty of building conducting robots will wear off. However, this seminar is meant to be a prototype for multidisciplinary courses that bring together students majoring in different areas. Our preliminary results show that such settings are successful in teaching students computational thinking and creative problem-solving. Future work includes collecting and analyzing data to support these findings.

7. ACKNOWLEDGMENTS

The authors would like to thank the orchestra of The College of New Jersey and its conductor Dr. Philip Tate for their collaboration and support. This work was supported by the National Science Foundation under Grant No. 0855973.

8. REFERENCES


