Work in Progress: Introduction of Failure Analysis to a First-year Robotics Course

Dr. Kathleen A. Harper, Ohio State University

Kathleen A. Harper is a senior lecturer in the Department of Engineering Education at The Ohio State University. She received her M. S. in physics and B. S. in electrical engineering and applied physics from Case Western Reserve University, and her Ph. D. in physics from The Ohio State University. She has been on the staff of Ohio State’s University Center for the Advancement of Teaching, in addition to teaching in both the physics and engineering education departments. She is currently a member of the ASEE Board of Directors’ Advisory Committee on P-12 Engineering Education.

Dr. Richard J. Freuler, Ohio State University

Richard J. (Rick) Freuler is a Professor of Practice and the Director for the Fundamentals of Engineering for Honors (FEH) Program in Ohio State’s Department of Engineering Education in the College of Engineering. He teaches the two-semester FEH engineering course sequence and is active in engineering education research. He is also affiliated with the Mechanical and Aerospace Engineering Department and conducts scale model investigations of gas turbine installations for jet engine test cells and for marine and industrial applications of gas turbines at the Aerospace Research Center at Ohio State. Dr. Freuler earned his Bachelor of Aeronautical and Astronautical Engineering (1974), his B.S. in Computer and Information Science (1974), his M.S. in Aeronautical Engineering (1974), and his Ph.D. in Aeronautical and Astronautical Engineering (1991) all from The Ohio State University.

Laurel Corrigan, Parker School Hawaii

Lauren Corrigan is a high school teacher at Parker School Hawaii and a former lecturer in the Department of Engineering Education at The Ohio State University. She earned both her Bachelor’s and Master’s in Civil and Environmental Engineering from Ohio State. She has two years of industry experience as an environmental engineering consultant. Her responsibilities included solid waste design, construction quality assurance, and computer aided design in support of various environmental projects. At Ohio State, Lauren engaged in teaching and curriculum development within the First-Year Engineering Program. Her research interests included the retention and success of students in STEM fields, with a particular focus on under-represented populations.
Work in Progress: Introduction of Failure Analysis to a First-Year Robotics Course

Abstract

This work-in-progress paper describes the first implementation of a failure analysis component added to an existing first-year cornerstone project course. The first-year engineering program at The Ohio State University provides honors students with the opportunity to engage in an intensive design-and-build robotics project. The primary educational objective of this course is to give students a realistic engineering experience, so that at the end of their first year, they can make educated decisions about whether engineering is the profession they want for themselves, and, if so, what particular engineering discipline they want as a major. To that end, the project includes many aspects of real-world engineering, including teamwork, budgeting, planning a project schedule, communicating orally and in writing, documenting, programming a microcontroller, constructing and wiring a device, and, of course, designing, testing, and refining of a product.

The robot project was first conceived over two decades ago, and it continues to evolve both technically and pedagogically. In the spring of 2017, one refinement was adding a failure analysis component to the course. It has always been part of the course that teams are required to participate in performance tests at several points during the term to determine whether their product is progressing according to schedule and executing as intended. The additional element required any team that scored fifty percent or less on a performance test to engage in a post-performance test analysis. They were to identify the causes for why the robot did not achieve the goals of the test, along with likely strategies for remedying the problems identified.

In the first offering of the course with this requirement, there were four performance tests, and about half of the participating teams engaged in one or more failure analyses. This paper describes the common causes students identified for their failures, as well as the range of solutions they proposed for fixing them. Additionally, a question on the course-end survey solicited feedback from the students regarding the educational value of the post-performance test failure analysis. Student responses were mixed, but have suggested refinements to the assignment for future offerings of the course.

Background and Rationale

Mistakes are part of the learning process. Educators recognize this, but students often struggle with the notion and as a result, many of them miss out on valuable learning opportunities. In a midterm exam study, a survey was given to 285 freshmen engineering majors and only 25% of students reported trying to learn from their mistakes while the material was fresh in their minds. The majority of the students put the test away and often never looked at it again. In another anonymous survey of 456 first-year engineering students, only 21% reported that they would use the exam again later, and many specified that would only be if the final were cumulative [1].
This data prompted the First-Year Engineering Honors Program at The Ohio State University (OSU) to implement exam corrections as a mandatory assignment for any student scoring less than 90% on an exam. It has been used for the last seven years, and the format is very similar to that described by Henderson and Harper in their introductory physics courses [2]. Instead of the instructor going over the test when it is returned, students have a few days to correct any mistakes, explain any errors, and submit this work as part of a homework assignment. This assignment is graded largely based on effort and the entire class reviews the exam, if necessary, only after submitting the exam corrections assignment. The goal is to help students use their exams in a formative way and learn from their mistakes. Henderson and Harper describe several small investigations that consistently show educational value in such an activity [2].

The same logic behind the exam corrections assignment was applied to another form of assessment in the First-Year Engineering Honors robotics course at OSU during the 2017 spring semester. During this class, students work in teams to design and build an autonomous robot to complete various tasks on a robotics course [3]. The design schedule for this project includes regular robot performance tests to help prepare students for both individual and head-to-head competitions near the end of the semester.

**The (Dreaded) Performance Tests**

Throughout this design, build, and test robot project, teams are evaluated on their robot’s performance regularly. At several points during the project, the teams are asked to complete one assigned task defined in the robot scenario on the course during a performance test. The main goal of these performance tests is to ensure continual progress throughout the project, especially on some of the more challenging tasks. As students approach the end of the project, they will have completed four different performance tests that cover all the objectives. This helps ensure that each team has demonstrated or achieved a baseline of performance in regards to each task. An overview of the performance tests for the 2017 robot competition is in Appendix A. The scenario for that year was that robots were assisting with tasks required for scientific research near a volcano.

Performance tests focus primarily on a single task each week. Student can choose to develop new robot control code each week or build on the code from previous tests. Teams who make their code more modular tend to have an easier time in programming the code for each new performance test, and they also have an easier time when all of the tasks must be combined later in the project. As the design of the robot progresses, there are often new mechanical and electrical or sensor changes made each week, in addition to the software changes. Thus each performance test often involves new or modified software and hardware.

The performance tests are a graded component of the class. Each of these tests is broken into sub-components with associated points. Each team is allowed up to five official attempts per performance test, and the highest score attained counts toward the course grade. To be considered official, the attempt must be observed and scored by a designated member of the instructional team and may only be completed during a class period. While a performance test may be completed in a class prior to the deadline, the deadline is firm on the day of the test.
Additionally, there are a stretch bonus and early completion bonus. Both of these opportunities are provided to encourage teams to move beyond the weekly expectations and complete the assigned tasks in a timely fashion. The early completion bonus provides incentive for teams to get ahead of schedule. The stretch bonus is intended to encourage teams to look at the bigger picture than just that week’s test and think about how the various elements of the challenge fit together. As such, the stretch bonus often involves navigating to the location of another task on the course. The instructional team is able to evaluate how each team is doing and devote specific attention to teams that may be struggling. This ensures that the majority of teams are usually seeing success when they reach the individual competition, the first time at which they are evaluated on whether their robots can complete the entire course or not.

Each performance test is graded out of twenty points and is typically broken down into four subtasks at five points apiece. The two bonus opportunities can add a total of five additional points to the score. The specific point breakdown for the performance test that is the focus of the analysis in this paper will be shared in the discussion that follows. Each test contributes 1.5% to the final course grade.

Even with a relatively small impact on the grades, students take the performance tests very seriously. In recent years, instructors observed that students placed a great deal of pressure on themselves to execute each performance test perfectly, but when a given performance test did not go as planned, students rarely spent time discussing with their group why their robot had failed. Due to the lack of analysis, teams would instead focus on the next performance test while staying with the failed design, or they would change their design without carefully thinking through their change. Some teams would approach each performance test with such tunnel vision that they would make major alterations to their robot each week, without thinking about how they would eventually need to perform all of the tasks together. These observations prompted the creation of the post-performance test assignment during the spring semester of 2017. Similar to the exam corrections, this assignment was constructed with the goal being to help students use the performance test in a formative way. Failure is a natural part of the engineering design process, and it is important for students to look at it that way, as opposed to something that should be dreaded and feared. As Lottero-Perdue and Parry state, “Practicing engineers acknowledge failure as a normal and expected outcome as part of the iterative nature of designing solutions to problems, although the end goal is that the solution …is not intended to fail.” [4] It is also important to note that students must learn to approach iteration as something more intentional than blind trial and error. Failures must be analyzed before adjustments are made to the design.

Details of the Failure Analysis Assignment and Overall Impressions

The failure analysis assignment is presented in Figure 1 below. Each post performance reflection is worth ten points, or slightly less than 1% of a student’s final grade. This number of points is consistent with other assignments throughout the semester that require a similar amount of work. Generally speaking, these second-semester honors students tend to take all assignments seriously, as they are always concerned about their grades. To determine whether students engaged in the kind of reflection and planning that was intended, the post-performance submissions from four of the nine course sections were collected and analyzed. Each of these sections had nine teams of four, for a total of 144 students on 36 teams. All of these teams did
well enough that they did not have to submit analyses for the first two performance tests, and only two teams were required to do an analysis for performance test four. This pattern was consistent with the rest of the course sections, as more than half of the teams fared poorly on the third test, but passed the others, often with bonus points. Therefore, the analysis will focus exclusively on the responses to the third performance test.

![Figure 1. Post-performance test assignment](image)

What was it about the third performance test that resulted in this distinction? The instructional staff knew in advance that it would be the most challenging, as it involved moving an unusually shaped object from one location on the course to another, incorporated a decision task, and required multiple forms of navigational input to be successful. The point breakdown for the performance test is shown below in Figure 2.

<table>
<thead>
<tr>
<th>Performance Test 3:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Touch core sample – 5 points</td>
</tr>
<tr>
<td>Remove core sample – 5 points</td>
</tr>
<tr>
<td>Deposit core sample in a bin – 5 points</td>
</tr>
<tr>
<td>Deposit in the correct bin – 5 points</td>
</tr>
<tr>
<td>Bonus - touch the antenna or its housing</td>
</tr>
</tbody>
</table>

![Figure 2. Point distribution for performance test 3](image)

In Figure 3 below, a CAD representation of the robot's area of operation on the test course is shown. For this performance test, the robot was required to navigate from the starting position (1) up a ramp (2) to the side of the volcano (3), extract a “core sample,” carry it to the lower level, read a light in the floor (4), and deposit the sample in one of two bins (5), depending upon the color of the light. The object in the left-hand bin in the figure is the core sample.
Additionally, while the robot had the option of navigating based upon data from the course’s robot positioning system (RPS) [5] on most of the course, the area near the core extraction site was an RPS “dead zone,” requiring the use of other navigational techniques to get to the core.

The core consisted of a 3-D printed handle attached to a large-diameter dowel rod by a metal screw. Teams developed a variety of approaches for extracting and holding it during transport. Some used a vice-like approach, gripping the top and bottom of the handle and using friction to hold it in place. Others incorporated some kind of teeth or other feature to wrap around the handle. Another popular option was using magnets to attract the screw. The methods for depositing it included holding it over the bin and releasing it, lowering it into the bin and letting go, and scraping it off on the side of the bin.

A total of twenty post-performance test three analyses were submitted in these four course sections, and they were roughly equally distributed, with four, five, or six turned in from each section. All of the submissions were written; none were video. The level of detail in the writing varied through the sample. Generally speaking, the stronger submissions tended to be well-organized, beginning with a general statement of the failed performance and an identification of the general areas of the issues, followed by more details on the issues. The proposed solutions that came after were linked to the problems that were identified. However, not all of the analyses were structured as logically or written as completely, and some of their shortcomings will emerge in the discussion below. A couple of them read more like detailed descriptions of the difficulties experienced in the days or hours leading up to the test, rather than an analysis of what happened during the test itself.
Qualitative Coding and Analysis

The failure analyses were coded qualitatively from a phenomenological interpretivist perspective [6, 7], since the data and conclusions drawn from them are the products of the environment in which the students and researchers were working. The qualitative coding was combined with a semi-quantitative analysis to describe trends in the content of the student submissions. The coding focused on the reasons provided for the failure and the planned approaches for solving the problems that had been identified, and 284 elements were coded. In general, much more of the writing was devoted to describing the issues that had been identified. Roughly 70% of the recorded codes were about problems, and only about 30% were about how they would be solved.

The trends that emerged in coding the problems identified by the teams are summarized in Table 1, with descriptions of the categories following. It is perhaps not surprising that many of the codes relate to the prompting questions in the assignment.

<table>
<thead>
<tr>
<th>Problem Area</th>
<th># of Teams Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time Management</td>
<td>19</td>
</tr>
<tr>
<td>Hardware</td>
<td>18</td>
</tr>
<tr>
<td>Software</td>
<td>16</td>
</tr>
<tr>
<td>Navigation</td>
<td>15</td>
</tr>
<tr>
<td>Testing</td>
<td>9</td>
</tr>
<tr>
<td>Team Issues</td>
<td>9</td>
</tr>
<tr>
<td>Making Major Change</td>
<td>5</td>
</tr>
<tr>
<td>Design</td>
<td>3</td>
</tr>
<tr>
<td>Interrelation</td>
<td>3</td>
</tr>
<tr>
<td>External Factors</td>
<td>3</td>
</tr>
<tr>
<td>Hubris</td>
<td>3</td>
</tr>
<tr>
<td>Equipment</td>
<td>2</td>
</tr>
</tbody>
</table>

Time Management: There was a reference to not planning time well, not spending as much time on the task as should have been, or not having enough time to accomplish the goal. A few representative comments include, “The mechanism we had designed to secure the lava core was not physically built until Tuesday night the week of the performance test,” “…we were behind schedule,” and “The cause of this problem was us not having enough time to properly program the robot…”

Hardware: Teams stated that hardware was a contributing factor to the failure or identified a specific hardware component as lacking. For instance, on one team’s robot, “the arm did not properly grip the core sample, and the gear that controlled the arm fell off.” Another found that “the magnet attached to the arm was not protruding enough to stick to the core sample; it was too flush with the end of the arm.”
Software: Teams either described an aspect of the software needing improvement or said in
general that software was an issue. One example was, “Using hard coding and RPS causes small
navigational errors to stack up and led to the robot to missing the core consistently.” [sic] Some
also described how the software was not properly controlling mechanisms on the robot.

Navigation: Teams either stated that their robot had difficulty navigating successfully or
described navigating mistakes. For instance, “…the turns were not always the 90 degrees they
were supposed to be. That caused the robot to be in a position slightly off from the wanted spot
in front of the core.”

Testing: The reflection stated that the team did not test enough or did not test effectively. One
team discussed at length how they delayed testing for several days until the core extraction
mechanism was completely constructed and only realized after the fact that they could have been
refining their navigation during this time. Another determined that if they had tested more, they
would have realized that they needed to modify their depositing strategy.

Team Issues: The reflection described problems related to how the team distributed the
workload for the task or communicated about their progress/approach to the performance test. It
should be noted that none of the responses described major arguments, disagreements, or
personality conflicts. In one case, a team described how they had given one team member sole
responsibility for the code, and while that had been a good approach prior to this test, it gave him
more of a challenge than any of them had expected. Another team had almost the opposite issue;
they found that for some reason they all focused entirely on building the robot’s arm for
controlling the core, and no one worried about coding or navigation until the test was imminent.

Making Major Change: The team was making a major change to at least one aspect of their
robot that hindered their progress toward the week’s test. One team wrote, “With only two days
to spare….we completely got rid of the core mechanism in the front and moved the omni-wheel
back to its original position.” And another: “If we were not busy rebuilding the robot (including
the chassis and drivetrain) from scratch and from scrap, we would have had plenty of time to
perfect the execution of our mechanism.”

Design: It was specifically stated that the design was flawed.

Interrelation: The team recognized in their reflection that the third performance test was more
complex than those that preceded, often referring to the many interrelated actions that all had to
be successfully accomplished to succeed. As one team stated, “If one of these [sub]tasks wasn’t
accurately completed, it often resulted in the other ones unable to be completed….we were able
to accurately complete each step multiple times. The team was just unable to piece it all together
on the course at that time.”

External Factors: The team felt that the failure was partially due to circumstances beyond the
team’s control, such as the availability of a part or the illness of a team member. One team had
ordered a 3-D printed part, but “there was a mistake made when our order was processed and it
was never made.”
Hubris: One team summarized nicely what was an issue for several: “This was a difficult task all around and we underestimated it.”

Equipment: Teams cited issues with equipment they had not designed. This was usually the fact that the battery on the robot controller had reached too low of a charge for reliable performance. In the two instances where this was mentioned, it was unclear whether it was due to lack of planning or overzealous testing, so this was kept separate from those categories.

Similarly, trends were identified in the solutions proposed to remedy the failure. These are summarized in Table 2.

<table>
<thead>
<tr>
<th>Solution Area</th>
<th># of Teams Reporting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software</td>
<td>11</td>
</tr>
<tr>
<td>Hardware</td>
<td>10</td>
</tr>
<tr>
<td>Time Management</td>
<td>10</td>
</tr>
<tr>
<td>Team Issues</td>
<td>9</td>
</tr>
<tr>
<td>Navigation</td>
<td>7</td>
</tr>
<tr>
<td>Design Modification</td>
<td>6</td>
</tr>
<tr>
<td>Strategy Modification</td>
<td>5</td>
</tr>
<tr>
<td>Equipment</td>
<td>1</td>
</tr>
<tr>
<td>Design Change</td>
<td>1</td>
</tr>
</tbody>
</table>

The five most frequently occurring categories refer to the same major areas of concern described above, but three new categories emerged that describe the approach planned for improving the robot’s performance:

Design Modification: Teams described what were judged to be relatively minor modifications to the robot’s existing code or structure, such as adding sensors or adjusting a few details of a function.

Strategy Modification: Teams described what appeared to be a rather minor modification to the strategy they were taking in approaching the tasks, either in the way they were attempting a particular task, or in the order of events. Many of the proposed modifications involved adding methods of navigation to the existing approach, for instance, incorporating line following to using RPS or bumping into walls.

Design Change: One team described two areas for “major overhauls” to their robot.

Digging deeper into the data to look for relationships between the reported problems and anticipated solutions yields an interesting set of observations. While the top five categories of problems and top five categories of solutions have the same elements, the orders are not the same. Also, it can be seen that solutions were suggested far fewer times than problem areas were
mentioned. Teams were quick to note what the problem appeared to be, but much less inclined to suggest how the identified problem was going to be attacked.

Perhaps more interesting was the number of teams which identified an issue but did not list any proposed solution to address it. Among all the teams, eight mentioned not enough time, nine mentioned hardware, seven mentioned software, and eight mentioned navigation as problem areas each with no potential solution being provided for any of the problems cited. In one particular case, hardware issues were referred to six times as a problem area with zero solutions being suggested. Additionally, a number of teams offered a solution in some areas where a problem had not been explicitly mentioned. Both of these circumstances reveal opportunities to improve the actual assignment instructions regarding the relationship between problems encountered and solutions proposed.

Assessment and Feedback

One big question is whether engaging the students in failure analysis led to stronger final designs. However, this is not a question that can be answered definitively. It is true that the 2017 final robot competition was the most successful in the history of the program, with more teams obtaining perfect scores than ever before. However, there were several aspects of the course that were adjusted in the hope of accomplishing just that, including a small reduction in the scope of the project. Therefore, while the failure analysis could have been a contributing factor, it is not known how much it was responsible for the students’ achievement.

To gather feedback from the students, two questions related to the assignment were included on the end-of-course evaluation:

- Did your team have to submit a post-performance test analysis?
- If yes, did you find the reflection to be helpful, either to the robot’s further development, or to your learning?

Of the 324 students taking the course, 289 completed the end-of-course survey, or 89%. Of those respondents, 57% reported that their teams submitted an analysis at some point during the term. This is essentially the same as the percentage of teams that did, 59%. As this data is anonymous, it is unknown how many of the respondents are from the same teams involved in the earlier qualitative analysis. It is also unknown how many different teams were represented by the respondents.

The perceptions of those whose teams did the assignment were mixed, with 51% stating it was useful, 44% finding it not useful, and 5% indicating that they were unsure. Many students that found the assignment to be helpful described the experience as something that helped the team communicate, saying that it helped them to “centralize the team’s thinking” or “look at the problems that occurred and analyze them more carefully.” Some used it to frame conversations with their instructional team, and others referred back to the post-performance write-ups when writing their final documentation. One person said it helped him realize that he needed to take writing the code more seriously. There was variation in how helpful they found it; while some credited the reflection with helping their teams discuss new ideas, others found it to be more of a
formalization of what they were already thinking: “Although we mostly knew what went wrong...the reflection probably helped us verbalize exactly what our issue was and how we could fix it. It wasn’t necessary, as we knew what went wrong, but it was a good assignment to make sure our team was on the same page.”

Those who did not find the experience helpful were more varied in their reasoning, but a fairly common element throughout was the implication that the causes of their failures were minor (“we...just needed another hour” or “…it was a mechanical error that caused the issue for the most part” or “we already knew what went wrong and what we wanted to fix”). In a few cases, the students believed the failure was not their fault (“...it was because of the store not our team. The store didn’t have an essential part we needed so we were unable to fully test.” [sic]) A few students reported not being able to determine what was wrong with their robots and so did not find the assignment beneficial. One particular comment aligned with the previous perceptions of instructors that the students’ “tunnel vision” sometimes prevented them from seeing the bigger picture: “No. We already knew what the issue was and how we were going to try to fix it. It did not add anything to the development of the robot. It may have been beneficial to my understanding of how engineering works in the real world but it didn’t help what was happening in the class.”

Summary and Future Plans

The first iteration of the failure analysis added to an existing robotics design-and-build course required students who did not earn more than half of the available points on a weekly performance test to reflect on what caused their design to fail and how they might remedy it. About half of the teams needed to complete at least one failure analysis. Analyzing the student work showed that they included more descriptions of the problems they encountered than the solutions they hoped to implement. A further exploration showed that part of this was because teams often did not actually discuss how they were going to attack the problems identified. Most of the analyses addressed some combination of hardware, software, navigation, time management, and team issues. While end-of-course evaluations showed that a simple majority of students found value in the assignment, they also confirmed that many students missed the point of the assignment.

These findings indicate that first-year students are capable of engaging in meaningful failure analysis, but that they need slightly more guidance than the original version of the assignment provided. In future offerings of the course, a modified version of the assignment will ask teams to make more explicit connections between the problems identified and the solutions proposed. The analyses employed to arrive at the results described in this paper will be repeated to see if the second iteration results in more coherent failure analyses and if the students see more value in the experience.

References

Appendix A: Overview of 2017 Performance Tests

2/24 – Performance Test 1 – Seismograph
Robot must navigate to and hold the seismograph button for at least 5 seconds.
Stretch bonus: Navigate to and touch part of the lava lever or its housing.

3/3 – Performance Test 2 – Lava Lever
Robot must navigate to the lava lever and toggle it such that the lever remains in the toggled position after the robot has lost contact with the lava lever and its housing.
Stretch bonus: Navigate to the light, read it, and display its color

3/10 – Performance Test 3 – Core Sample
Robot must navigate to the core sample, remove it, and deposit it in the correct bin.
Stretch bonus: Navigate to and touch the antenna or its housing.

3/24 – Performance Test 4 – Antenna and Stop
Robot must navigate to the antenna, turn it 90 degrees, return to the bunker, and push the stop button.
Stretch bonus: Navigate completely to the upper level and back down.