Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #46867

An Aeroelasticity Experiential Learning Activity with AI-Driven Comparative Analysis and Evaluation

Mr. Bobby F Hodgkinson, University of Colorado Boulder

Bobby Hodgkinson is an instructor in the Smead Aerospace Engineering Sciences and has 6 years experience with the Aerospace Engineering hands-on curriculum at CU Boulder. His focus is in hands-on curriculum instruction, and the development and implementation of activities to enhance experiential learning.

An Aeroelasticity Experiential Learning Activity with AI-Driven Comparative Analysis and Evaluation

Bobby Hodgkinson
hodgkinr@colorado.edu
Smead Aerospace Engineering
University of Colorado Boulder

Abstract

This paper introduces an aeroelasticity lab activity designed for sophomore aerospace engineering students, combining aerodynamics and structural mechanics. The lab aims to bridge the gap between theoretical knowledge and real-world applications by guiding students through industry-relevant problems, offering hands-on experience in model creation, data acquisition, and model validation. Students are tasked with creating aerodynamic loading and structural deformation models for an aircraft wing and its spar beam. Using a low-speed wind tunnel, they collect data to compare against their models, fostering essential industry skills like model validation, experimental testing, and comparison between the two—skills difficult to impart in a traditional lecture setting.

To further enhance this experiential learning activity, we have incorporated AI-based tools into the lab, specifically chatbot generated solutions for the lab's core problems. In a novel extension of the assignment, students critically evaluate the AI-generated solutions by comparing them to their own analytical work, investigating the correctness of the AI generated solutions. This AI-assisted comparative analysis encourages deeper engagement with both AI outputs and traditional engineering methods, fostering critical evaluation and explanation skills.

This integration of AI generated solutions into the lab is designed to expose students to the strengths and limitations of readily available AI tools, preparing them to interact with these technologies as they become more prevalent in engineering practice. Students inherently learn to recognize areas where AI excels, as well as areas where human expertise remains critical. By maintaining existing lab infrastructure while updating the assignments with AI tools, the lab remains both modern and adaptable to future technological advancements. Additionally, this framework enables instructors to easily update assignments semester to semester while maintaining the core activities.

A key objective of this adaptation is to prepare students for a future where AI-generated solutions may surpass even the best human abilities. However, a skill that remains irreplaceable is the ability to critically assess the correctness of solutions—whether human or AI-generated. This paper presents findings in the form of student reflections on this modern adaptation of

comparative analysis.

1 Introduction

Aeroelasticity is a field in aerospace engineering combining aerodynamics and structural mechanics to understand the interaction between aerodynamic forces and structural responses. At the University of Colorado Boulder, a sophomore-level Aerospace Sciences Lab introduces students to these concepts through an experiential learning framework. The course's aeroelasticity lab forms a core component of this pedagogical approach, combining theoretical and hands-on elements to prepare students for real-world engineering challenges.

This paper discusses a lab activity combining aerodynamics and structural dynamics for sophomore aerospace engineering students. The lab activity's primary aim is to help students bridge the gap between theoretical knowledge and real-world applications by guiding them through a problem similar to one they may be tasked with in industry. The goal of introducing this activity was that its hands-on nature would both further students' knowledge of the associated course material as well as introduce them to valuable industry skills that they would not have otherwise obtained in lecture.

Specifically, the lab activity tasks students with experimentally obtaining data to create aerodynamic loading and structural deformation models for an aircraft wing and its spar beam. After creating these models, students collect loading and deformation data using a low-speed wind tunnel and compare the results to their models. Through this process, the lab activity aims to provide students with experience creating and validating models as well as data acquisition and testing—skills that are widely used in the aerospace industry.

In recent years, artificial intelligence (AI) tools have become increasingly interesting in the field of engineering and in education. Recognizing this, the lab integrates generative AI-based tools, specifically generative AI based solutions, as an exploratory exercise for students. These tools serve as both a benchmark and a critical evaluation exercise for students. By comparing their own solutions with AI-generated ones, students gain insights into the strengths and limitations of generative AI while enhancing their critical thinking and problem-solving skills. This also serves as an informative study on student perceptions on integrating generative AI into their own works. The reflective nature of this exercise can give student's insight into their own usage of generative AI in other courses and as they evolve into an engineering career. Broadly speaking, this integration explores a modernized approach to engineering education, aligning with industry trends and preparing students for the evolving technological landscape with generative AI at the forefront.

The approach of providing AI-generated solutions to students is akin to the educational concepts of *worked examples* and *comparative analysis* which have been employed for several decades. The integration of worked examples with self-explanation strategies emerges as a powerful approach for enhancing learning and skill acquisition across various disciplines[1,2]. Worked examples provide learners with detailed, step-by-step solutions, reducing cognitive load and enabling a more accessible path to understanding complex concepts[1,3]. This is particularly beneficial in fields like computer programming, where the abstract nature of the subject matter can pose significant challenges for novices[1]. Moreover, these examples help students to track problem-solving steps and understand the underlying rules, often proving more effective than lengthy textual explanations[1,3]. The effectiveness of worked examples is further amplified when coupled with self-explanation, a process where learners actively engage in clarifying concepts for themselves, leading to a deeper understanding and improved metacognitive skills[1,2,4]. This strategy encourages learners to connect new information with their prior knowledge, promoting knowledge generation and a more profound comprehension of the material[2].

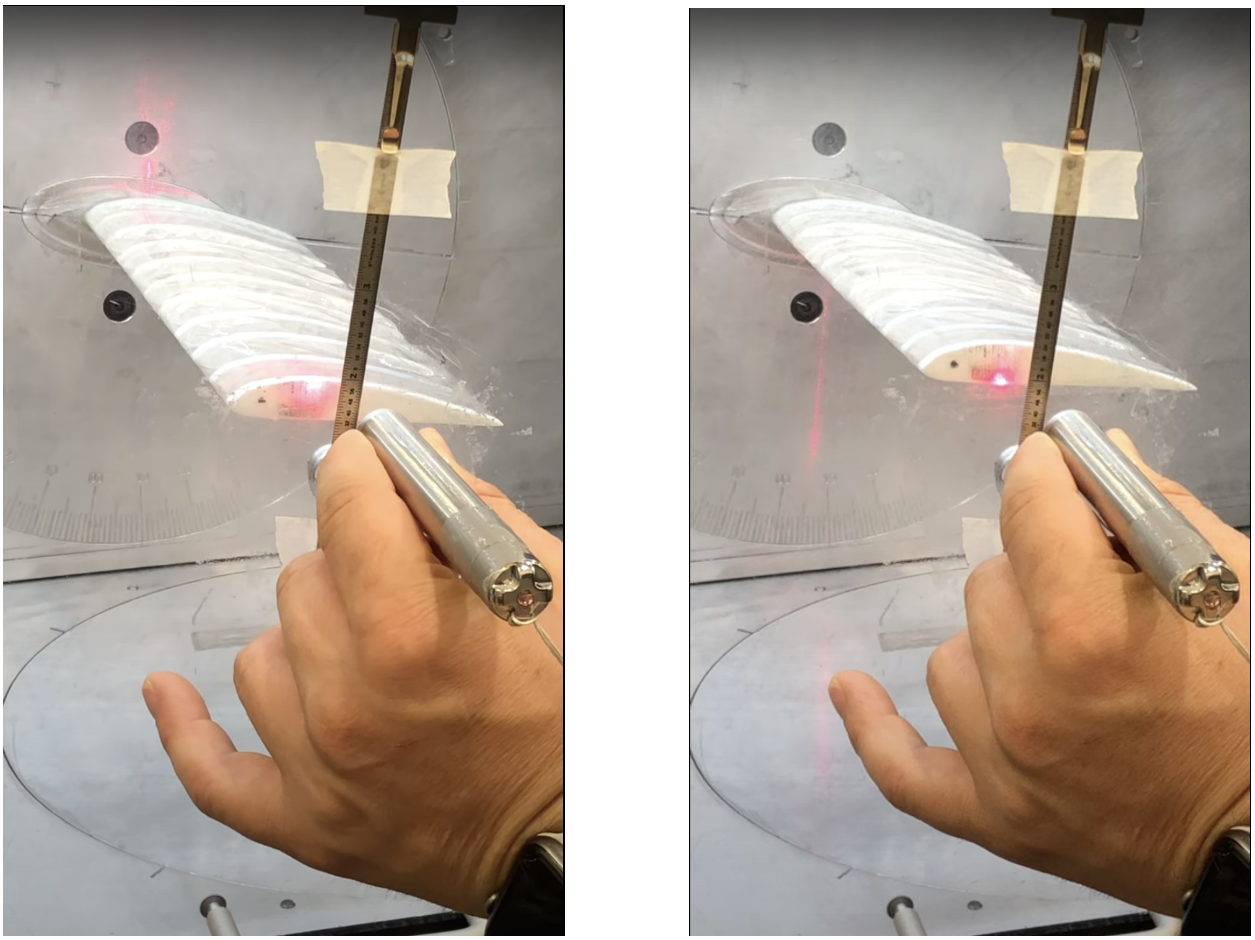
Comparative analysis, another key educational technique, can be facilitated through the use of worked examples and self-explanation. By exploring different approaches to problem-solving and comparing them, students can gain a more nuanced understanding of the subject matter[2]. This approach aligns with the principles of active learning, which encourages student involvement and hands-on experiences[5]. Furthermore, reverse engineering, which involves deducing input information from observed output, complements comparative analysis by promoting skills in planning, observation, and abstraction[5]. Students who engage with worked examples and are encouraged to self-explain benefit from active exploration of the material, which allows them to connect individual components to the overarching goal of the problem, enhancing their familiarity with specific syntax, and identifying effective strategies for algorithm design[2]. This method is particularly useful in computational contexts, where understanding the "why" behind code is as crucial as the "how"[2]. In addition, AI tools, such as ChatGPT, can be used as an educational resource to support learning and research, but educators need to be proficient in their use to integrate them effectively[6,3]. However, AI cannot replace key higher order skills, as was shown when analyzing AI-generated laboratory reports in chemistry, which highlighted several deficiencies, such as inability to maintain consistency, generate references, and suggest experimental errors[3].

In the realm of computational thinking, algorithmic explanations can serve as a powerful means of instruction. These explanations, when constructed by students, serve as a way of solidifying their understanding of computational practices and the underlying concepts[7]. This approach also promotes a deeper understanding of scientific processes, by connecting them to the language of computing and allowing students to use algorithmic concepts[7]. In-code commenting serves as a useful way for students to self-explain their understanding of programming code[2]. By using in-code comments as a self-explanation activity, students engage in an active exploration of the worked examples and connect individual lines of code with the overall goal of the problem, and identify effective strategies for algorithm design[2]. Therefore, worked examples should be carefully designed with comments that promote self-explanation, because this will help students in the development of their programming skills.

In summary, the reviewed literature supports the use of worked examples and comparative analysis, especially when integrated with self-explanation strategies, as effective methods for learning and skill development. These techniques, when combined with active learning, reverse engineering, and the judicious use of AI tools, create powerful learning opportunities that promote deeper understanding and enhance problem-solving capabilities across various disciplines[5,6]. This approach also helps to support the development of computational literacy, especially when integrating algorithmic explanations[7].

2 Methods: Experiment Overview

The aeroelasticity lab comprises three milestones designed to progressively introduce students to aerodynamic and structural principles necessary to analytically predict the deflection of a simple wing and compare the prediction to experimental results. The activity culminates with deflection measurements of a 10-inch long, 3.5-inch chord cantilevered wing in a 12 inch x 12 inch x 24 inch deep test section of an educational windtunnel. Figures 1a and 1b show the non-deflected and deflected wing respectively. The wing is set at a 10° angle of attack and and airspeed of 30 $\frac{m}{s}$ which results in approximately 10 mm of deflection. A stationary laser pointer is held to provide a visual indication of the deflection.

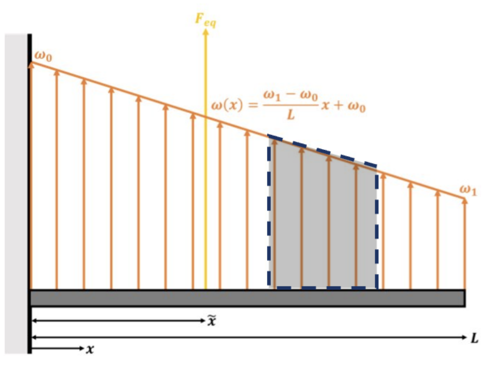


(a) No airflow, no deflection

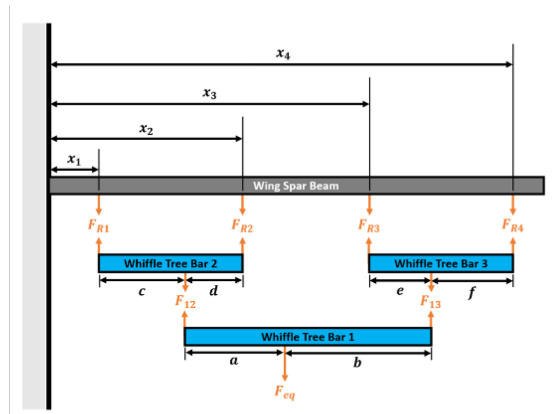
(b) 30 m/s airflow, 10 mm deflection

Figure 1: Comparison of deflection with and without airflow.

In the first milestone, students familiarize themselves with aerodynamics and wind tunnel testing principles alongside an introduction to the wiffletrees. Wiffetrees are useful test setups for cantilevered beams and aircraft wings[8,9,10,11]. At this stage in their education, students are only becoming familiar with two dimensional aerodynamics so they do not yet possess an understanding of the complicated nature of lift on a finite wing. With a basic conceptual discussion, students can begin to appreciate that the lift distribution and hence the distributed load may be approximated by a rectangle, triangle, or trapezoid shape. This allows the structural analysis to predict the deflection of the wing spar as a cantilevered beam subject to simple geometric loads. At the same time, students explore discretization of a load to design the structural wiffle tree for future tests. They also build MATLAB functions to divide a provided loading shape into four sections which will then translate to the physical dimensions of a wiffle tree experiment to experimentally test the deflection of a simply supported beam. Diagrams for the analysis of the wiffle tree are provided in Figure 2. The wiffle tree experiment is shown in Figure 3 for context.



(a) Trapezoidal loading approximation for wiffle tree analysis showing one of four discrete sections



(b) Diagram of wiffle tree to simulate a distributed load

Figure 2: Wiffle tree analytical diagrams

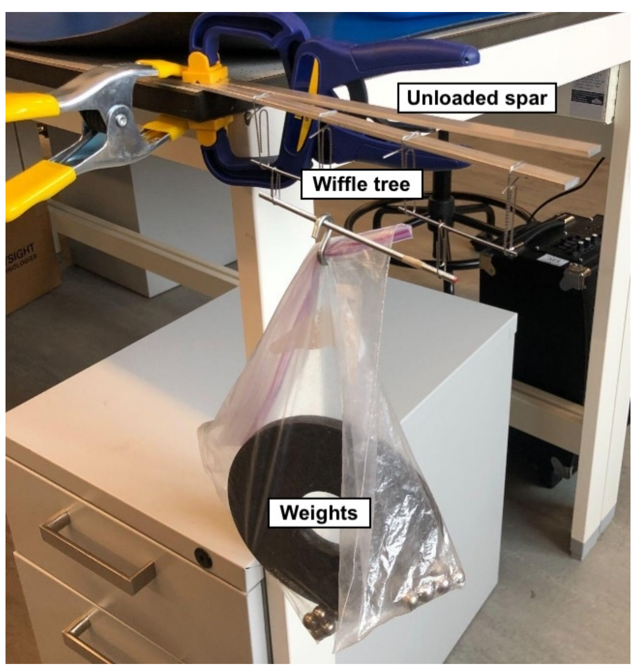


Figure 3: Constructed wiffle tree to simulate a distributed load on a cantilevered beam.

In the first milestone activity, students also estimate airspeed using a Pitot-static probe in a simple desktop wind tunnel by applying Bernoulli's principle and the concept of differential airspeed. Figure 4 shows the desktop wind tunnel. This simplified experiment helps students better grasp the relationship between differential pressure and airspeed. This relationship is a foundational concept for aerodynamics and allows students to progress to the next milestone where they will relate differential pressure to lift and drag.

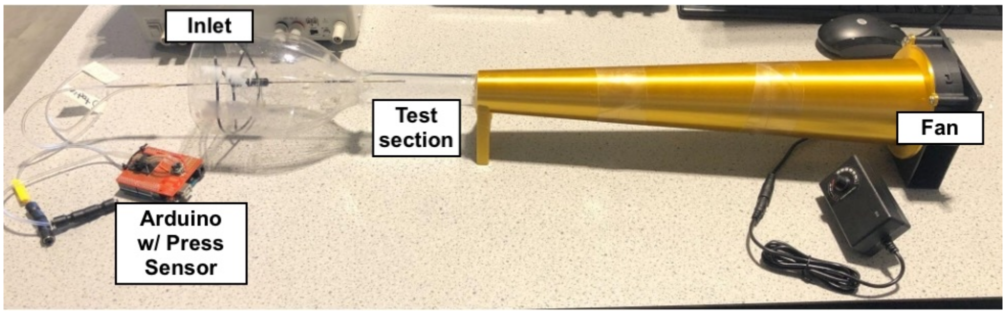


Figure 4: Desktop wind tunnel made with a 2-liter plastic bottle inlet. A traversing Pitot-static probe made with two hyperdermic needles is connected to a differential pressure sensor with data displayed on a nearby monitor. Airflow moves from left to right. Total length of tunnel is approximately 2 feet.

In the second milestone, students build on these foundations by analyzing the lift and drag characteristics of a cambered airfoil. They consider factors such as angle of attack and pressure distribution. Students collect experimental data using a cambered airfoil at different angles of attack in a low speed wind tunnel. A 3D printed airfoil with 16 small holes around the surface is shown in Figure 5. These 16 small holes are connected through a series of tubes to a 16-channel differential pressure scanner. These readings allow for the estimation of the pressure distribution around the wing leading to an approximation for the lift and pressure drag.

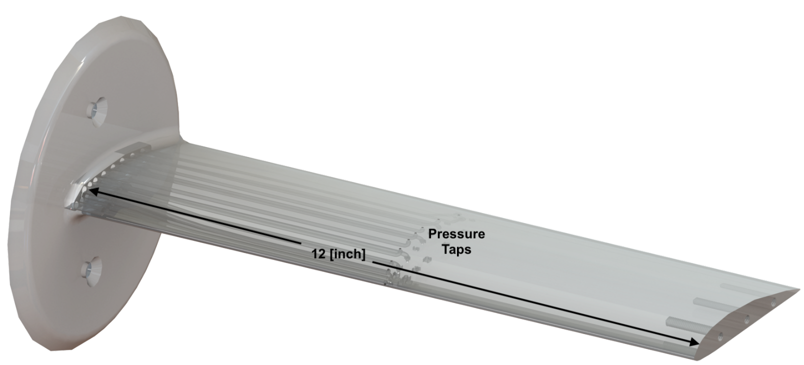


Figure 5: 3D printed model of a Clark Y14 airfoil with 16 pressure taps for lift and pressure drag experimentation.

A comparison to NACA data[12] using a similar airfoil are made to help student's gain confidence in their analysis. At the same time student's focusing on the structural analysis aspects, derive expected beam deflection equations under various load distributions, namely rectangular, triangular, and trapezoidal patterns. Approximate loading values are provided to the structural team from the aerodynamic team's analysis of the wind tunnel experiment. This milestone bridges aerodynamic insights with structural mechanics, enhancing students' ability to relate complex concepts.

The third and final milestone synthesizes findings from the previous stages by examining the spanwise lift distributions and comparing analytical predictions with experimental results. Student's analyze pressure distribution data from a finite wing with 16 pressure taps at 10 different spanwise locations from the root to the wingtip. Figure 6 shows the 3D model of this wing. Students are provided with the test data for this milestone due to time constraints in the course.

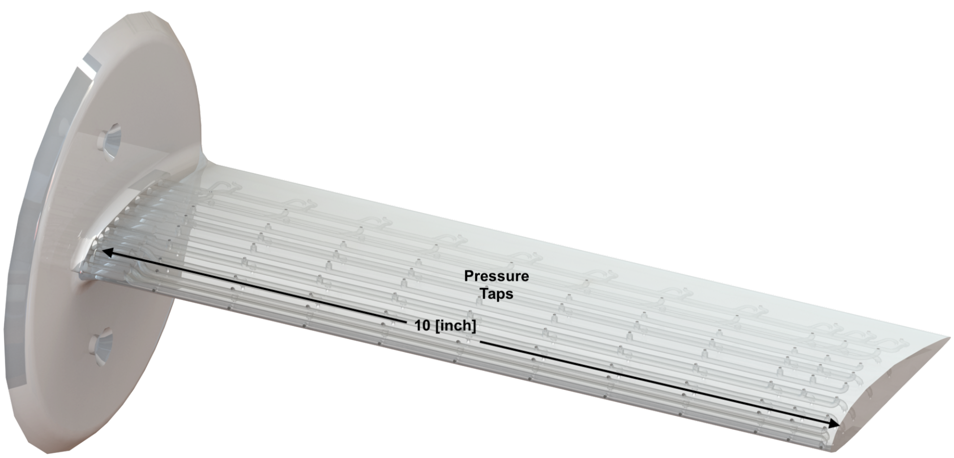


Figure 6: 3D printed model of a Clark Y14 airfoil with 16 chordwise pressure taps at 10 different spanwise locations.

This data is used to determine an estimate for the lift distribution shape and approximate values for the finite wing. This aerodynamic analysis provides the structural team with an experimentally informed loading estimate to inform their beam bending models from earlier, and to allow them to determine more representative values for the wiffle tree experimental test. Students construct the simple wiffle tree shown in Figure 3 using these values. Additionally, students observe the finite wing in the wind tunnel as shown in Figure 1 to measure the deflection of the wing spar subject to aerodynamic loading. This allows students to draw comparisons of beam deflection using their analytically derived model(s), the experimentally informed wiffle tree structural test, as well as the deflection of a wing in a wind tunnel. This milestone reinforces collaborative skills as students integrate aerodynamic and structural findings into a cohesive analysis. This approach also exposes students to the principles of analytical models, experimentally informed models, experimental observations, and the relationship between theory and experimentation.

3 Methods: Generative AI Comparative Analysis

Artificial Intelligence (AI) tools were incorporated into the lab as an exploratory pedagogical element. For each milestone, students were provided with generative AI solutions addressing key challenges such as load discretization and lift coefficient calculations. It is worth noting that the term "*solution*" is used here to indicate *a means of solving a problem* and not to indicate that these AI solutions were correct. These solutions, created using frontier generative AI models (specifically chatGPT 4o[13] and o1-preview[14]), offered an additional perspective on problem-solving approaches. To ensure meaningful engagement with AI tools without compromising the aerodynamics and structural learning objectives, students compared the provided AI solutions to their analytical models, identified discrepancies, and justified the correctness or inaccuracies of the AI solution. They also reflected on the AI's utility and limitations in the context of engineering tasks. This approach was designed to foster critical evaluation skills and an understanding of AI's potential role in engineering workflows. The AI generated solutions were created by prompting ChatGPT 4o and o1-preview using the student facing lab assignment(s) and zero-shot prompts which were essentially copied from the original assignment document. In other words, while the course instructor provided students with the AI-generated solutions, students could have crafted prompts on their own using only the lab assignments (see Figure 7 in appendix A for a sample prompt). The AI solutions were not modified to be more or less correct as a key objective of this exercise was to evaluate the benefits and limitations of readily available AI generated solutions. We elected to provide a consistent set of AI generated solutions to the students as an attempt to standardize the experience. This approach also seeks to eliminate any disparities that could arise from differences in student access to AI tools. By providing a consistent set of AI-generated solutions, we aimed to ensure that students who had access to more advanced or paid AI models, such as ChatGPT Pro, did not have an unfair advantage over those using free-tier models or other AI tools. This standardization sought to maintain a level playing field while still allowing students to participate in the activity regardless of their familiarity with generative AI.

Student performance and engagement were assessed through deliverables for each milestone, which included MATLAB scripts, plots, and written analyses. While studies on the efficacy of this lab activity on student performance in the overall aerospace curriculum are valuable, the focus of this paper is on quality assessment/quality improvement (QA/QI), interesting results involving AI-generated solutions, and the comparative analysis therein. Reflections encouraged students to evaluate the AI tools, focusing on areas where AI insights were valuable or misleading. Students were also directed to reflect on their overall sentiment in terms of using AI as an assistant in their school work and the potential for generative AI in their future careers as engineers. The actual reflection assignment is provided as Figure 8 in the appendices (see appendix B). It is worth noting that students were given the flexibility to provide reflections in formats other than the one suggested, and with more or less content than specified. This approach was intended to encourage creative participation in the activity.

4 Results

The integration of AI tools into the aeroelasticity lab provides a unique opportunity to assess generative AI's impact on student learning, engagement, and problem-solving skills. Student reflections revealed both strengths and limitations of the AI-generated solutions, offering insights into how these tools may be used in engineering education. These reflections were awarded as extra credit opportunities so as not to overly burden and potentially distract from the main learning objectives of the course. Voluntary responses were received from 97 students in a class of 315. It is important to note that this may indicate a preferential bias in the results to align more with students who are in favor of using generative AI than those who are opposed. An attempt to remedy this bias was made by informing students that credit would be awarded solely based on their completion of the reflection and not on their impressions or even their findings of using generative AI in this format. These responses were analyzed collectively using a combination of instructor review, natural language processing using deidentified submissions to search for common themes, and also NotebookLM[15] to gain additional insight on the corpus of reflections. The survey's free-form, reflective natured emphasizes qualitative insights, with student quotations serving as the primary contribution to the results section. Since the reflection assignment did not include a structured set of standardized questions, it was not intended for a rigorous, quantitative analysis (e.g., 'X percent of respondents indicated...'). An attempt to extract quantitative data from this format would risk being inappropriate and potentially misleading. This investigation was intended as a quality assurance (QA) and quality improvement (QI) effort, setting the stage for future research where a more rigorous, quantitative approach could be employed under proper research protocols.

The following sections highlight common themes identified from the corpus of student reflections. To ensure readability and maintain the conciseness of the paper, representative quotes illustrating these themes are included in the respective sections of the appendix (see appendix C). The quotes are from submissions by students who gave explicit permission to use anonymized quotes from their submission to aid in: improvements to the curriculum, proposals, and/or publications.

4.1 Strategies for Comparing Results

Students employed a variety of strategies to compare their results with the AI-generated outputs. Direct comparison of numerical results was a common first step, where students analyzed values such as maximum airspeed, deflections, stresses, and strains. They also examined intermediate calculations to ensure consistency. Visual inspection of graphs proved particularly valuable, as students compared trends and shapes in lift and drag coefficients, pressure coefficients, and deflection curves. Overlaying their graphs with the AI's outputs often highlighted key differences and areas for further investigation. See: appendix C.1.1.

Code analysis played a significant role in understanding the AI's methodologies. Students dissected the AI-generated code to evaluate its structure, variable declarations, and overall approach. This allowed them to identify discrepancies in constants and methodologies. Logical checks and hand calculations were also crucial tools, enabling students to validate results against fundamental principles and physical expectations. Statistical metrics, such as percentage deviation, were occasionally used to quantify the differences between AI and student outputs. See appendix C.1.2.

4.2 Areas of Alignment

Students found alignment in several areas, particularly in general trends and shapes of graphs, which were often consistent between the AI and their own results. Numerical agreement was also noted in initial calculations, such as equivalent force estimations. The AI's core logic and use of general equations frequently aligned with student approaches, offering reassurance in shared methodologies. In some instances, results were nearly identical, with only minor numerical discrepancies due to rounding or differences in implementation. See appendix C.2.

4.3 Areas of Difference

Discrepancies were more pronounced in specific numerical values, such as uncertainty calculations and deflection values. These differences often stemmed from the AI's hardcoded assumptions or incorrect partial derivatives. Data processing methods varied, with the AI struggling to clean and normalize data effectively. Differences in error analysis approaches were also evident, as the AI relied on MATLAB functions while students sometimes performed manual calculations. See appendix C.3.1.

Code implementation diverged significantly, with the AI using distinct methodologies for functions, variable declarations, and coding styles. Graphing inconsistencies, such as missing elements or incorrect normalization, further highlighted the AI's limitations. Additionally, the AI occasionally produced inaccurate or generalized responses that lacked the specificity required for the lab's tasks. See appendix C.3.2.

4.4 Reconciling Differences

To address these differences, students revisited equations and assumptions, double-checked boundary conditions, and carefully analyzed the AI's code. Correcting errors in both their own work and the AI's outputs often resolved discrepancies. Lab documentation and hand calculations served as valuable references, helping students validate their approaches. In some cases, students chose to trust their own methods over the AI's outputs, particularly when their findings aligned with physical expectations and lab guidelines. See appendix C.4.

4.5 Impact on Student Learning

The exercise of evaluating AI-generated solutions appeared to enhance students' understanding of core concepts. Comparing results deepened their grasp of structural mechanics and aerodynamic principles, reinforcing the significance of boundary conditions and load distributions. The AI's novel approaches often prompted students to explore alternative methodologies, broadening their problem-solving skills. Debugging and error analysis further honed their critical thinking, as students identified and addressed discrepancies between their work and the AI's outputs. See appendix C.5.1.

However, the AI's oversimplifications and occasional inaccuracies posed challenges. Some students felt hindered by the AI's lack of intermediate steps or reliance on predefined formulas. Inconsistent responses and incorrect calculations occasionally caused confusion, requiring additional effort to resolve. See appendix C.5.2.

4.6 Increased Confidence in Assessing External Tools

The experience improved students' confidence in evaluating external tools and methodologies. By scrutinizing the AI's assumptions and outputs, students developed critical assessment skills and learned to identify potential errors. They recognized the importance of foundational knowledge in validating AI-generated results and gained insights into the strengths and limitations of AI tools. These skills will be essential as AI becomes an integral part of engineering workflows. See appendix C.6.

4.7 Future Implications

The findings suggest that AI tools have significant potential to enhance engineering education when used thoughtfully. As students noted, the ability to critically assess AI outputs will be crucial for future engineers navigating an AI-driven industry. By engaging with AI tools in a reflective environment, students are better prepared to leverage these technologies effectively while maintaining their role as critical thinkers and decision-makers. See appendix C.7.

Discussion

The integration of generative AI into the aeroelasticity lab was aimed at enriching the student learning experience while seeking to explore the broader implications of AI in engineering education. Through the feedback and reflections gathered from students, several key insights emerged regarding the reception of this pedagogical strategy, the lessons learned, and potential implications for the future.

Overall, students expressed mixed but largely positive sentiments toward the integration of AI-generated solutions in the lab activities. Many recognized the value of these tools in enhancing their understanding of complex concepts, with several students noting that comparing their own solutions to solutions generated by AI deepened their grasp of fundamental principles such as load distribution and boundary conditions. As one student remarked, "Evaluating [the AI's] derivations deepened my understanding of core concepts."

However, challenges were also noted. Some students expressed frustration with the AI's oversimplifications or its reliance on predefined functions, which occasionally hindered their ability to follow the logic or intermediate steps. These observations underscore the dual nature of AI as both an assistant and a potential source of confusion in educational contexts. It was observed that students who lacked a strong foundational understanding struggled more with assessing the AI's outputs, reinforcing the importance of preparing students to critically engage with AI tools without forming an over-reliance on them.

This exercise uncovered the potential of AI tools to serve as effective educational aids, provided they are implemented thoughtfully. Students noted that working with AI helped them develop critical evaluation skills and confidence in assessing external tools. These abilities are likely to be essential for future engineers, who will work in environments where AI plays a significant role in decision-making processes.

Moreover, the activity highlighted the benefits of showing students a way to view AI not as a replacement for their expertise, but as a complementary resource. Several students reflected the sentiment that, "AI complements our work but cannot replace us." This shift in perspective—from seeing AI as an answer key to viewing it as a collaborator—is a helpful step in preparing students to thrive in an AI-driven industry.

The integration of AI tools into this lab activity offers valuable lessons for educators in other disciplines seeking to modernize their curricula. The comparative analysis exercise proved effective in helping students identify both the strengths and limitations of AI as well as to help improve their own conceptual mastery of the underlying activities. This approach is widely applicable beyond engineering, offering a framework for educators in diverse fields to incorporate generative AI into their teaching.

By engaging students in activities that require them to validate AI-generated outputs, educators can cultivate critical thinking and problem-solving skills that are increasingly important in the modern workforce. Such exercises also help students build resilience and adaptability, positioning them as valuable assets to future employers and proactive members of society as AI technology continues to evolve.

As generative AI becomes more prevalent, the ability to work alongside these tools will be a defining characteristic of successful professionals. The lab's integration of AI not only enhanced students' technical skills but also provided a foundation for understanding how to effectively leverage AI in their future careers. Students learned to identify errors, reconcile discrepancies, and evaluate the credibility of AI-generated solutions—skills that will remain critical as AI continues to shape the engineering landscape.

By exposing students to the benefits and deficiencies of AI, this exercise also encouraged them to think critically about their role in an AI-driven future. Rather than viewing AI as a threat to their employability, students began to see it as a tool that, when used effectively, can enhance their value to future employers. This mindset shift is essential for fostering a generation of engineers who are not only proficient in their technical domains but also adept at navigating the complexities of an AI-augmented workplace.

The integration of generative AI into the aeroelasticity lab was an informative step toward modernizing engineering education and preparing students for the challenges of an AI-driven future. By fostering critical evaluation skills and encouraging students to engage deeply with both human and AI-generated solutions, this activity laid a foundation for continued exploration of how AI can enhance learning across disciplines. The insights gained from this exercise will inform future efforts to integrate AI into educational activities, ensuring that students are well-equipped to work alongside these tools as they advance in their careers.

References

[1] R. Alhassan. The effect of employing self-explanation strategy with worked examples on acquiring computer programing skills. *Journal of Education and Practice*, 8:186–196, 2017.

[2] C. Vieira, A. Magana, A. Roy, and M. Falk. Student explanations in the context of computational science and engineering education. *Cognition and Instruction*, 37:201–231, 2019. doi: 10.1080/07370008.2018.1539738.

[3] Joseph K. West, Jeanne L. Franz, Sara M. Hein, Hannah R. Leverentz-Culp, Jonathon F. Mauser, Emily F. Ruff, and Jennifer M. Zemke. An analysis of ai-generated laboratory reports across the chemistry curriculum and student perceptions. *Journal of Chemical Education*, 100(11):4097–4106, 2023. doi: 10.1021/acs.jchemed.3c00581.

[4] Cristina Conati and Kurt VanLehn. An intelligent computer tutor to support self-explanation while learning from examples. In *International conference on intelligent tutoring systems*, pages 399–408. Springer, 2000.

[5] H. Elizalde, I. Rivera-Solorio, Y. Pérez, D. Guerra, and R.A. Ramírez-Mendoza. Educational framework based on collaborative reverse engineering and active learning. *International Journal of Engineering Education*, 25 (5):1063–1071, 2009.

[6] Alberto Grájeda, Johnny Burgos, Pamela Córdova, and Alberto Sanjinés. Assessing student-perceived impact of using artificial intelligence tools: Construction of a synthetic index of application in higher education. *Cogent Education*, 11(1):2287917, 2024. ISSN 2331-186X. doi: 10.1080/2331186X.2023.2287917.

[7] Amanda Peel, Troy D. Sadler, and Patricia Friedrichsen. Algorithmic explanations: an unplugged instructional approach to integrate science and computational thinking. *Journal of Science Education and Technology*, pages 428–441, May 2022. doi: 10.1007/s10956-022-09965-0. URL https://doi.org/10.1007/s10956-022-09965-0.

[8] NASA. Lessons learned from large-scale aerospace structural testing. Technical report, NASA, 2021. URL https://ntrs.nasa.gov/api/citations/20205011135/downloads/SDM2021_Lesons-Learned_2021_final-o.pdf.

[9] Yusuf Giri Wijaya Muksin Nur Mufidatul Ula, Aryandi Marta. Wing static test of lsu-02 ngld aircraft using whiffletree method. *AIP Conference Proceedings*, 2088(1):030025, 2019. URL https://pubs.aip.org/aip/acp/article/2088/1/030025/798077/Wing-static-test-of-LSU-02-NGLD-aircraft-using.

[10] Kitplanes Staff. Testing, testing! *Kitplanes Magazine*, 2021. URL https://www.kitplanes.com/testing-testing/.

[11] Joseph T. Callihan. F/a-18e/f full scale structural fatigue testing. In *Proceedings of the Aircraft Structural Integrity Program Conference*, 2007. URL https://asipcon.com/proceedings/proceedings_2007/2007_PDFs/Thurs_0330_Callihan_abstr.pdf.

[12] Robert M. Pinkerton and Harry Greenberg. Aerodynamic characteristics of a large number of airfoils tested in the variable-density wind tunnel. Technical Report 628, National Advisory Committee for Aeronautics (NACA), 1938. URL https://ntrs.nasa.gov/api/citations/19930090937/downloads/19930090937.pdf.

[13] OpenAI. *ChatGPT 4o*, 2024. URL https://openai.com/research/chatgpt. Generative AI model developed by OpenAI.

[14] OpenAI. *ChatGPT o1-preview*, 2024. URL https://openai.com/research/chatgpt. o1-preview version of ChatGPT, developed by OpenAI.

[15] Google. Notebooklm, 2024. URL https://research.google/notebooklm/.

A Sample Prompt

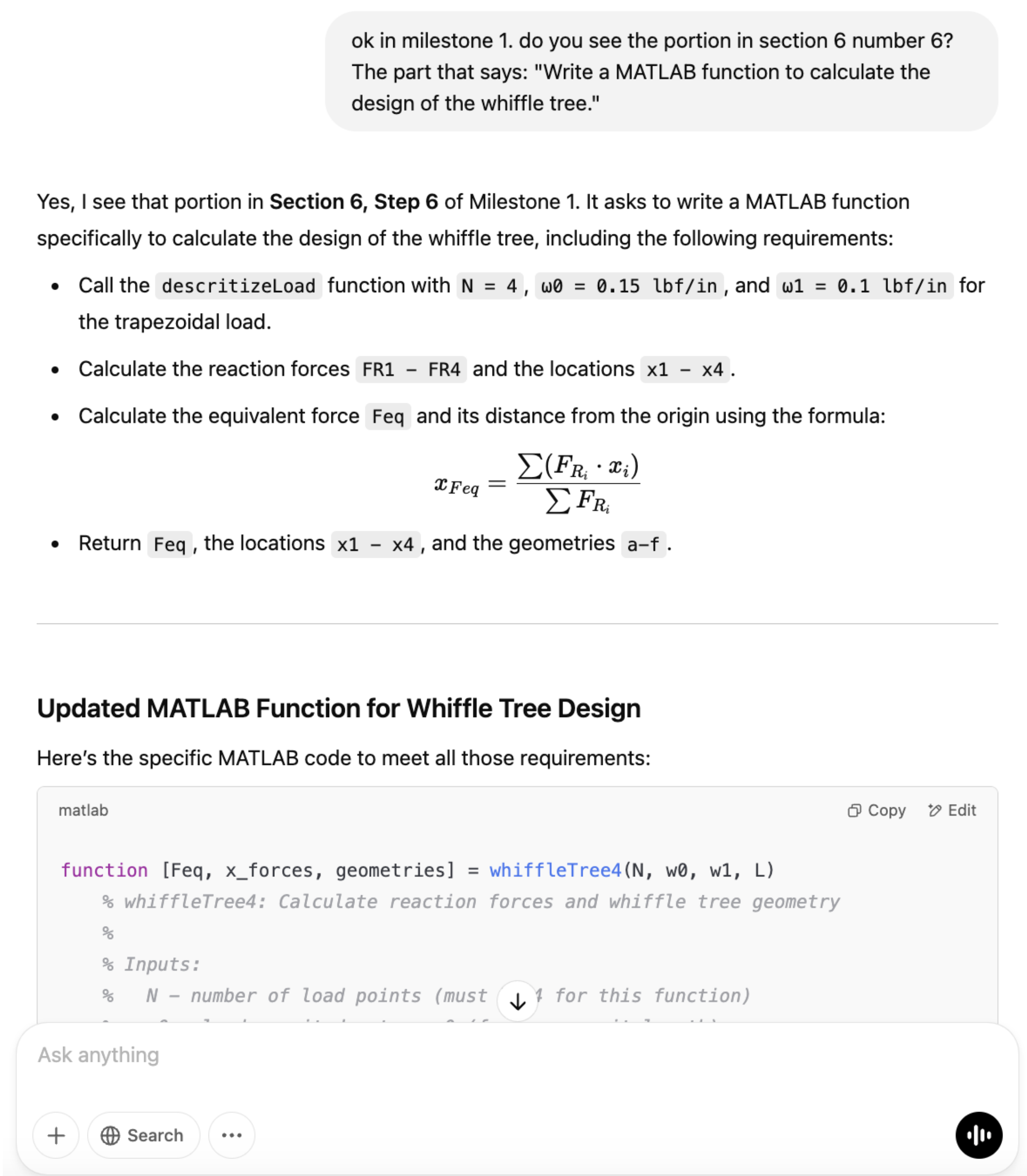


Figure 7: Sample prompt. Milestone 1 was uploaded as a .pdf prior to this prompt.

B Assignment

Reflections on AI Program Helpfulness for Improved Engineering (RA.I.phie) /ræl.fi/

1 AI Integration in Analysis

1. How did you approach integrating the AI-generated results into your workflow?
   - Did you use the AI's results as a starting point, a comparison tool, or a secondary validation? Why?
   - Were there aspects of the AI's analysis that you found particularly insightful or novel? Which ones, and why?
2. What strategies did you use to compare your own results to the AI's outputs?
   - Were there areas where your results aligned with the AI's? Where did they differ?
   - How did you reconcile differences between your analysis and the AI's findings?
3. Did the AI's outputs provide any new perspectives or insights that you might not have considered on your own?

2 Reflection on Learning and Understanding

1. Did the process of evaluating the AI's analysis deepen your understanding of the core concepts? If so, how?
   - Was there any aspect of the exercise where you felt the AI hindered rather than enhanced your learning? Explain.
   - Do you feel more confident in your ability to critically assess external tools or methods after this exercise?

3 Preparing for the Future

AI's Future Role in Engineering

1. Do you foresee AI tools like this being a natural part of your workflow as a future engineer? Why or why not?
   - Based on this experience, what skills do you think engineers of the future will need to effectively leverage AI tools?
   - If AI tools become ubiquitous in engineering, how do you think that will change the way engineers approach problem-solving and design?
2. How do you think AI might redefine what it means to be an engineer in the future?
3. What do you think is the most valuable lesson you've learned from working alongside AI in this exercise?

Figure 8: Assignment Screenshot

C Student Quotes

C.1 Strategies for Comparing Results

C.1.1

1. “For a specific spanwise position (200in), the AI script calculated a lift coefficient of 0.765CL = 0.765, while our script produced 0.758. Similarly, the AI script gave 0.0348 for drag coefficients, compared to our 0.0355. These small discrepancies arise from differences in data processing, such as averaging methods and trailing-edge pressure estimations. Additionally, our script incorporates more robust handling of port data and explicit interpolation for trailing-edge pressure, leading to increased reliability in the results.”
2. “The Aerodynamics script is built well and executes without error. The result values were very similar to our validated results. . . if they were found at all. Critical values included max airspeed, which was found as 17.6118 m/s (Student), and 17.5901 m/s (AI). The average airspeed was found to be 17.0113 (Student)”
3. “The results aligned in the initial discretization [sic] of aerodynamic loads and centroid calculations in Milestone 1. However, differences emerged in Milestone 2’s internal stress and strain computations, where my approach underestimated deflections compared to the AI’s outputs.”

C.1.2

1. “ I compared the AI-generated results to my own primarily through hand calculations and logical checks.”
2. “When comparing results with the AI, we compared our graphs, and if the graphs were different, then we added print statements in the code to find where the two diverged.”
3. “To compare the code, I initially started by comparing the outputs of my group’s code versus the AI’s code to find any immediate discrepancies. However, the AI often had mismatched units or strange values for its constants, so this wasn’t always immediately an effective comparison.”
4. “When comparing results with the AI, we compared our graphs, and if the graphs were different, then we added print statements in the code to find where the two diverged. The figures on the aerodynamics side are closely aligned with the AI for most graphs. However, the graphs for Cp were different from each other. We reconciled the differences between that graph by calculating some of the values at which our graphs differed by hand. In that case, we determined that we were right. The AI outputted graphs that showed the error analysis for CD and Cl, which is something that we wouldn’t have thought of before. Additionally, in milestone 3, it outputted a graph for Cp vs. normalized chord length, which we could have used to debug the code.”
5. “The AI code is much easier to follow, but our code shows how we went about solving the problem based on the way it was written.”

C.2 Areas of Alignment

1. “Overall comparison with our findings: Overall, the graphs match the shapes that our team produced, and the main objectives of Milestone 2 were completed. Some small values are inaccurate which disrupted small parts of each plot. Because the pressure coefficient was inaccurate at a point, the coefficients of lift and drag were affected. The AI missed some small details but produced the big-picture goals of Milestone 2. ”
2. “The results aligned in the initial discretization [sic] of aerodynamic loads and centroid calculations in Milestone 1. However, differences emerged in Milestone 2’s internal stress and strain computations ”
3. “The general equations utilized were the same as our work and AI-generated work, but AI was unable to calculate answers correctly, so we were unable to check our answers. AI also differed from our work in that it could not provide specified results; AI only provided us generalized responses, which were of little use to us. It also differed from our work by providing methods that we did not think of. ”

4. “The CD and CL vs. Angle of Attack graphs lined up really well. The values lined up to 5 decimal places, likely because the way that the AI calculated the coefficients of lift and drag (it used functions) did not depend on what it (wrongly) found for coefficient of pressure. It seems like all of the differences between our results and the AI’s results are where it made a small mistake like not normalizing the chord length or miscalculating the x value for the last point. ”

C.3 Areas of Alignment

C.3.1

1. “For the aero part of Milestone 1, the results after modifying the AI’s constants was the exact same for the maximum airspeed. However, for the related uncertainty, we did not obtain the same values. Our result was 0.135 m/s and AI’s was 0.0675 m/s. ”
2. “For Milestone 1 (structures), the AI code hinges on some erroneous assumptions. It assumes that the centroids for the three loading types are the same. This is not the case because the centroid of a triangle and rectangle are different as a triangle has a centroid a third of the way through the wide part and a rectangle has a centroid halfway in the middle. A trapezoid has a centroid that combines both of these properties. It is these centroids that determine the location of the point loads on the spar which vary depending on the load type and load parameters. Therefore, only the rectangular load calculations were the same because it is the only load where the centroids actually lie halfway through the width. ”
3. “ The AI had errors in finding the pressure coefficient, and the graphs for the error analysis included interpolated data that was not entirely accurate. It did not properly close the values for the pressure coefficient leading to a stray line on the graph. This affected some values for the coefficients of lift and drag.”

C.3.2

1. “In Milestone 2 for the Aero team, Ralphie AI took a much different approach for the code, utilizing massive functions that included most of the solving. Which is not the most effective way to code, this was evident by the time it took to run the two codes. ”
2. “Sometimes, it seems like the AI tried to reduce the number of lines of code by putting multiple commands on one line, which had mixed results as far as readability. ”
3. “The AI’s graph looked pretty good at first, but the AI added an extra seemingly random point, possibly due to incorrect indexing, and the AI did not normalize the x axis.”
4. “In some cases, the AI’s reliance on specific functions (e.g., trapz for integration) provided results without detailing intermediate steps, which slightly hindered the learning process.”

C.4 Areas of Alignment

1. “Differences were reconciled by revisiting the equations and assumptions used in the analysis. For example, discrepancies in the deflection were addressed by verifying the moment of inertia and modulus of elasticity values used in the calculations. ”
2. “When comparing results with the AI, we compared our graphs, and if the graphs were different, then we added print statements in the code to find where the two diverged. The figures on the aerodynamics side are closely aligned with the AI for most graphs. However, the graphs for Cp were different from each other. We reconciled the differences between that graph by calculating some of the values at which our graphs differed by hand. In that case, we determined that we were right. ”
3. “For the aero part of Milestone 1, we noticed that our partial derivatives were different from those the AI had created. We went over those and we believe that AI had them wrong, introducing a factor of ½ in the partial derivatives for both pressure and temperature, which is why we believe their uncertainty is half of what we got. Once we modified rA.I.phie’s partial derivatives, the uncertainty was the same as ours. ”

4. “When reconciling differences between the AI’s results and my own, I referred to the lab document, and generally went with my own results. I did this because I understood how I arrived at my results. ”

C.5 Impact on Student Learning

C.5.1

1. “The AI’s outputs didn’t introduce any new perspectives or insights that I hadn’t already encountered in lectures or considered on my own. However, evaluating its derivations deepened my understanding of core concepts. The step-by-step presentation, even if not entirely accurate, was invaluable in connecting equations and understanding their relationships.”
2. “The AI deepened our understanding of core concepts by offering different approaches and encouraging deeper analytical thinking. For instance, when discrepancies arose between our results and the AI’s, we were required to conduct a thorough analysis to identify errors, which significantly enhanced our grasp of these subjects.”
3. “The process of evaluating AI-derived analyses deepened my understanding of load distributions, especially the influence of varying loads (trapezoidal and triangular) on deflection and stress.”
4. “Absolutely, evaluating the AI’s analysis reinforced my grasp of structural mechanics and aerodynamic principles. It clarified the significance of boundary conditions and load distributions.”
5. “Analyzing the AI’s systematic breakdowns clarified the interdependence between shear force, bending moment, and deflection equations.”
6. “This process of investigating differences not only reinforced our understanding but also allowed us to ask new questions and discover insights we might not have considered independently.”

C.5.2

1. “In some cases, the AI’s oversimplifications of the complex problems felt like a hindrance, especially when deeper theoretical understanding was needed. I think AI is a great tool, but it’s up to you to firstly understand what it is doing. Without a solid understanding of the theory or methods, the AI will simply confuse you by oversimplifying.”
2. “In some cases, the AI’s reliance on specific functions (e.g., trapz for integration) provided results without detailing intermediate steps, which slightly hindered the learning process. Understanding how the AI reached certain conclusions required backtracking through the code.”
3. “When using AI-generated responses in comparison to our own work, we were given inconsistent responses. Specifically, when checking our code against an AI-generated code, we were given a generalized code that was actually not helpful at all because of its lack of personalization to our project. AI does an adequate job at generating code and evaluating our code, but it did not fully understand the parameters of our project, so it offered faulty advice at times.”

C.6 Increased Confidence

1. “This exercise has boosted my confidence in my ability to critically assess external tools such as AI. This is because I will be able to identify potential methods or simplifications that I have already seen in order to write proper code.”
2. “After working with AI in this assignment, I feel more familiar with reading code from AI and therefore more confident in using it to my advantage. Finally I am more experienced in reconciling discrepancies between AI and myself.”

3. “The exercise improved my ability to critically evaluate external tools by cross-referencing outputs, testing assumptions, and identifying gaps or ambiguities in solutions.”
4. “After this exercise, I feel substantially more prepared to examine the work of external tools. Going back and forth with my validated code and the results of the AI tool output has allowed me to see the strengths and weaknesses of AI. Particularly AI struggles with abstract problems and methods of problem-solving but excels at syntax and implementation once the method is found.”
5. “I feel much more confident in analyzing AI code since I know what to look for. General syntax and busy work such as plotting and variable declaration are nothing profound and the AI nails this every time. Where the AI needs to be evaluated is in the way it derives its equations. The AI is able to draw from a much larger pool of information than me which is how it arrives at the same results but with drastically different methods.”
6. “Using these tools and seeing how they responded to different questions, coding challenges, and mathematical calculations was very enlightening. They are NOT always correct, no matter how convincing their explanation may seem. One must be very conscious of what they are asking when putting it into an AI tool and must have a grasp on the concepts in order to check that it is correct. This was the case for the statics team, as it could not produce a correct whiffletree. We have a better understanding of where it typically goes wrong, and what its strengths (more basic coding) and weaknesses (too complicated of code and some scientific and mathematical explanations) are.”

C.7 Future Implications

1. “The most valuable lesson ive learned while looking at this AI model is to use AI in the future, but to keep an eye out if there is something out of the regular. Things such as graphs could help you to find these mistakes, as well as creating simple code to verify the results of the AI. by having the AI not only write the code but explain what the code does, it makes it easy for revisions to be made.”
2. “The most important lesson we learned was that AI can be wrong too. Thinking about AI as another group member instead of the answer key was something I needed to overcome. Changing my view on AI was essential in reconciling discrepancies between the two pieces of code.”
3. “The most valuable lesson I’ve learned from this exercise is that AI is not a one-size-fits-all solution. Human intuition remains essential in engineering. AI, while a powerful tool, cannot think, innovate, or match the creativity of the human mind. It complements our work but cannot replace us.”
4. “To effectively leverage AI, engineers must fully grasp core concepts and fundamental principles. These are the skills that distinguish humans from machines, providing the intuition and logical framework needed to ”fact-check” the errors and inconsistencies that AI may generate.”
5. “I believe that you need a firm understanding of the core concepts as a foundation to perform a successful analysis on an AI response. If you do not have a baseline understanding, how can you judge whether the AI is correct or not without blindly trusting it? I believe that AI can both hinder and enhance your learning, and it ultimately comes down to the student and how they are choosing to use the tool.”
6. “In the future, engineers will need to be effective at verifying the solutions produced by AI in order to work with it effectively. Additionally, engineers will need to find a way to test solutions produced by AI effectively. Most of the time, AI is right, but when AI turns out to misunderstand, we need to know when that happens so that we can correct it. Finally, engineers need to understand the code written by AI so that corrections can be made when needed.”