Title
Exploring Educational Needs and Practices in Structural Analysis

Authors
Lanning, Joel
Roberts, Matthew
Wiggins, Brandon

Publication Date
2024-06-01

DOI
10.18260/1-2--47406

The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #42559

Exploring Educational Needs and Practices in Structural Analysis

Dr. Joel Lanning, University of California, Irvine

Dr. Joel Lanning, an Associate Professor of Teaching at the University of California, Irvine, specializes in seismic design for civil structures, including bridges and buildings. His research revolves around advancing tools and techniques for improving the cyclic resilience of structural components. Dr. Lanning is passionate about teaching and is dedicated to developing strategies and tools for effective learning. His teaching philosophy emphasizes the creation of strong learning communities and the use of active learning methods to engage and challenge his students.

Dr. Matthew W Roberts, Southern Utah University

Dr. Roberts has been teaching structural engineering topics for 22 years. He is a professor of civil engineering at Southern Utah University.

Prof. Brandon K Wiggins, Southern Utah University

Prof. Wiggins is an Associate Professor of Physics at Southern Utah University. He works in computational astrophysics and specializes in large dataset analysis and visualization, machine learning and artificial intelligence, and high-performance computing.

Exploring Educational Needs and Practices in Structural Analysis

Abstract

For decades, the gap between academic training and practical skills in structural engineering has been a concern for both practitioners and educators. This disconnect is often attributed to several factors including an imbalance in the teaching of analytical and theoretical methods, too much or too little exposure to computer software, inadequate focus on developing an intuitive understanding of structures, and a deficiency in nurturing engineering thinking and imaginative problem-solving abilities.

This paper examines the alignment between industry needs and academic curricula for structural analysis education. Recent surveys of practitioners and educators reveal agreement on the importance of both classical methods and competency with analysis software. However, an investigation of structural analysis course descriptions in 264 U.S. undergraduate civil engineering programs indicates that only 54% explicitly cover computer techniques, while just 22% teach approximate methods useful for verifying software—a skill rated as 'Important' or 'Very Important' by the vast majority (>90%) of practitioners and educators surveyed. This highlights the disconnect between valued industry skills and current teaching practices. The investigation further reveals there may also be mismatches such as limited emphasis on topics like load path understanding, again despite its applied value. As automation shapes the profession, developing forward-thinking, integrated curricula that merges classical skills with software proficiency and an understanding of structural behavior is increasingly critical.

Introduction and Background

It should come as no surprise that there is a lack of consensus on exactly what to teach engineering students in order to properly prepare them for professional practice. The topics and skills required of new engineering graduates are broadly influenced by ABET through program accreditation and often propagated through tradition. But, when it comes to specific areas of practice such as structural engineering, or even more specifically in the topic of structural analysis, there are persistent challenges that educators face in order to prepare students to efficiently join the profession.

This problem is many decades old but the specifics have shifted along with advances in technology used in both engineering practice and education. A brief but comprehensive history of civil engineering education including the 18th and 19th centuries is given by Aparicio and Ruiz-Teran [1]. Civil engineering education in the U.S., starting around the late 18th century, followed two European traditions of British and French origins. The former placed emphasis on practical application of scientific principles, while the latter put more emphasis on sound theoretical understanding as a basis of engineering practice. However, many civil engineers were still trained through apprenticeships and so they received a great deal of practical training.

With the technological and economic advancements of the mid and late-19th century, together with the establishment of land grant universities, many engineering degree programs were developed and led to the decline of the apprenticeship model. In the early 20th century an acceleration of developing theoretical knowledge quickly forced both university models to take on less and less discipline-specific technical knowledge. It was then that the gap between industry needs and education began to widen [1].

The advent of World War II and the increase in funding for technical research had a profound impact on engineering education as curricula shifted towards math and science and away from "drawings and shopwork." [2] This move towards a more math- and science-based curriculum was perhaps most prevalent in electrical engineering, but other disciplines including civil engineering were also impacted.

In the 1960s, Malcolm Gregory, in [3] and [4], described the lack of "engineering attitude" in engineering education. He reflected on the 19th century apprenticeship model and early 20th century practical application training that nurtured an engineering mindset through personal contact and hands-on learning. Gregory believed past approaches better instilled design intuition and real-world know-how, developing design proficiency by imitation, learned rules-of-thumb, and learned the practical rules of design and construction.

Gregory described a problem with overly theoretical training. He argued the growing emphasis on analytical skills came at the expense of teaching engineering intuition and practical application. This caused students to view problems as just mathematical exercises, disconnected from real-world applicability. Gregory believed past educational models produced graduates better equipped to handle novel problems and transition to practice more easily.

Through the second half of the 20th century, though, the situation was complicated further with the introduction of mainframe computers, [5] and [6], then widespread access to personal computers [7]. A popular sentiment was expressed by Wagh, in 1987 [7], that likely still resonates with many today: *students ought to be taught fundamentals and practical aspects using manual calculation before using computer software tools.*

Hand Calculations or Analysis Software Competency?

Throughout the last several decades many have argued for increasing the emphasis on hand calculations in order to combat the over reliance on computers. As Powell points out in [8], many seasoned engineers and educators complain that students do not have a "feeling" for structures and that they often point to hand calculation methods like moment distribution as the solution.

Even in 2007, [9], at least a decade after ubiquitous use of computers in engineering practice became common, engineers were warning that the use (conflated as the over reliance) of computers deprived new engineers the experience that countless hours of calculations provided in order to gain a feeling for structures.

But is this 40-year-old attitude, that mastery of hand calculations is necessary to understand structures, really applicable for today's engineers who are vastly more computer competent than they were in the 1980s? Since then, the internet, cloud computing, BIM, and many other advances have occurred and been incorporated into practice for engineers to remain competitive. Further, these technologies are now simply native to how people work in the third decade of the 21st century. Certainly we should expect graduates to be fairly competent in the use of structural analysis and design software upon graduation.

Today more than ever graduates enter a profession that uses tools vastly more powerful than were available just a few years ago. Meanwhile, their structural engineering curriculum was likely minimally different from two or even three decades ago (or more!). In fact, some papers on this discrepancy are now even two decades old but read as if they were just published. Criswell in 2004 [10], discusses the dilemma that educators face in covering classical methods, such as the iterative moment distribution, in an environment where students have tools that instantly provide exact solutions. Criswell describes the situation twenty years ago:

> *"...increasingly, a primary task of our graduates in their role [as] young engineers is to be intelligent users and managers of design and analysis software. This change in the role of the individual designer… from being the direct producer of numerical solutions to the manager changed with utilizing software written by others to produce a correct numerical solution is a major shift, perhaps qualifying as a paradigm shift."*

As we approach the Artificial Intelligence (AI) paradigm shift, the intensity of the imbalance between academic training and professional needs is certain to accelerate especially with respect to the use of computer and classical methods. In fact, a recent five-part series of articles on automation in structural engineering was published in Structures Magazine, indicating the push for AI tools (starting with [11]). If there was ever a time to reevaluate and seriously consider modernizing how we, as educators, prepare students to enter the profession, it is now.

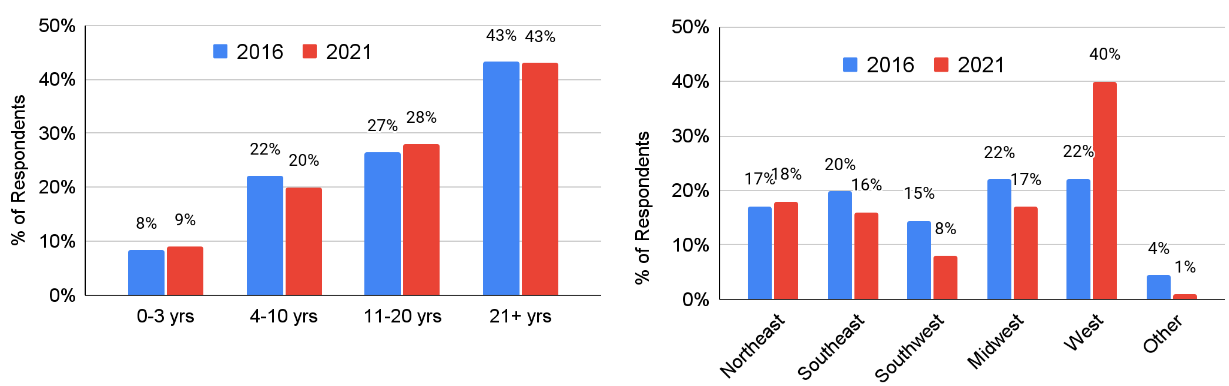
Therefore, this paper begins a larger effort by the authors to explore the needs of the industry regarding structural analysis skills and the current educational practices, with one major component being the balance of classical and computer methods. The ultimate goal is to provide a suggested forward thinking and compelling way of teaching engineers to understan structural behavior through analysis.

In this work, we summarize recent surveys of structural engineering practitioners and educators with an emphasis on the results most relevant to structural analysis topics and skills. In order to better understand how these opinions align with the current curricula in the U.S., course descriptions are analyzed from the 268 ABET-accredited U.S. undergraduate civil engineering programs.

Contemporary Opinions on Structural Analysis of Practitioners and Educators

The National Council of Structural Engineers Association (NCSEA) Basic Education Committee (BEC) recently conducted two wide-reaching surveys asking practitioners about skills and educational requirements they value in new hires. Structures Magazine has reported on the 2016 and 2021 survey results in [12] and [13]. Additional details from the 2021 survey results were provided by Dong and Francis in [14]. The NCSEA BEC also conducted an educator survey in 2019 focused on the structural engineering curriculum of 168 undergraduate programs, [15]. Here, only the survey data on structural analysis is reported and is compared in a way that it was not originally presented in order to draw conclusions more directly about structural analysis (and not structural design, technical communication skills, etc.). The survey was, in a way, granular on specific structural analysis topics and did not ask participants to rate a comprehensive list of particular analysis methods. It did ask for an importance rating of ancillary analysis topics like load path, stability, and dynamics. But, as it pertains specifically to structural analysis (i.e., moment/shear diagrams, deflection analysis, load distributions) the survey questions were more in the direction of how structural analysis education should be approached, e.g., the importance of classical methods, coverage of matrix methods, and the use of computer software. However, this aligned well with the takeaways from the literature review.

Approximately 415 practitioners participated in the 2016 survey and approximately 515 did so in 2021. Well-seasoned engineers with over 21 years of experience were very well represented in both years. The great majority (~70%, each) had at least 11 years of experience, as shown in **Fig. 1**. The majority of respondents worked primarily on buildings in both surveys.



(a) Experience Level (b) Region of Primary Practice

Figure 1 - 2016 and 2021 NCSEA Practitioner Survey Demographics, [12] and [13]

Structural Analysis Courses

Using the 2019 NCSEA Curriculum Survey, [15], the NCSEA BEC developed a suggested structural engineering curriculum, [16], which contained a detailed description of many courses including three structural analysis courses: Structural Analysis I - Determinate, Structural Analysis II - Indeterminate, and Structural Analysis III - Matrix Methods. Referencing these, practitioners in 2021 were asked to rate the importance of each to the undergrad curriculum. The results are compared in **Fig. 2** to corresponding course and/or topic descriptions appearing in the 2016 Practitioner survey. The rating scheme is presented in the figure.

It is no surprise that practitioners overwhelmingly strongly valued a full course covering determinate analysis and a full course covering indeterminate analysis. Note, in 2016 no distinction was made between determinate and indeterminate analysis courses. Matrix methods, on the other hand, received mixed importance ratings, with many (44%) in 2021 indicating students need a full course in matrix methods. Only 15% and 14%, in 2016 and 2021, respectively, indicated that matrix methods were "Not Important."

Skills and topics related to structural analysis were rated in the same way. In **Fig.** 3, both surveys indicate very strong support for loading and load paths. Structural stability and structural dynamics also were also strongly advised to be full courses within the curriculum. Only finite element analysis received a significant portion rating it as "Not Important" (19%).

Classical versus Computer Methods in Structural Analysis

A thematic analysis was performed, [12], on independent personal responses provided in the 2016 survey results. Many responses (=42%) included strong views regarding the need for classical or "hand" calculation structural analysis methods. Many (=38%) expressed the importance of the use of computer modeling and students' skills in interpreting results.

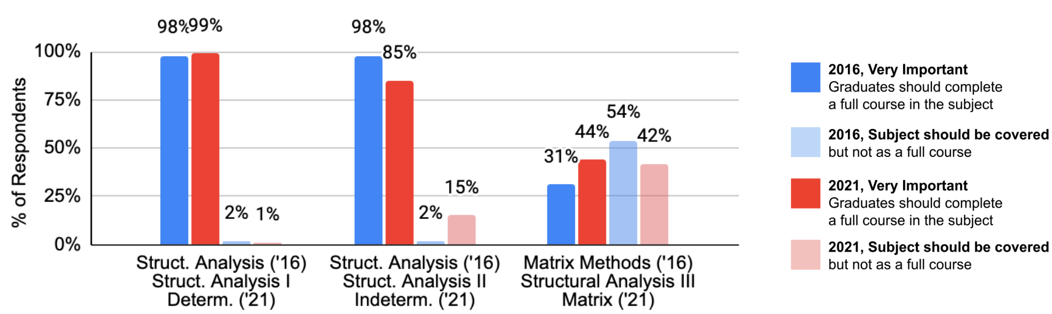


Figure 2 Practitioner importance rating of structural analysis courses 2016 vs. 2021, [12] and [13]

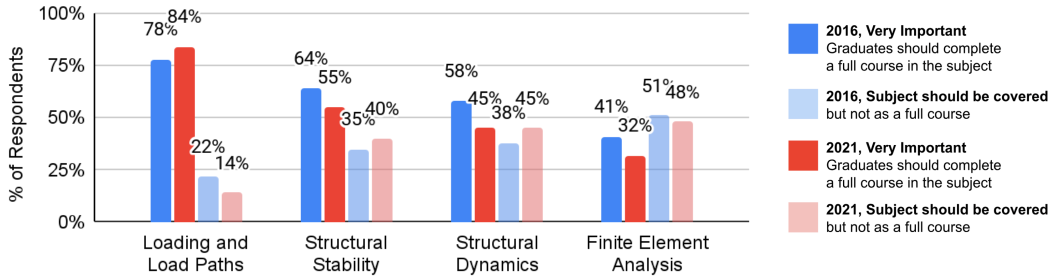


Figure 3 Practitioner importance rating of structural analysis related skills 2016 vs 2021, [12] and [13]

The 2019 curriculum survey [15] and the 2021 survey [14], then, contained explicit questions asking educators and practitioners, respectively, to rate the importance of classical and computer methods in the undergraduate curriculum. The results and rating choices are shown in **Fig. 4**. Both groups rated computer methods as "very important" at a higher rate than hand/classical methods. However, both groups recognized the importance of both hand/classical and computer methods. Interestingly, the "not important" rating was phrased in a way that placed importance on the other method. In other words, rating hand methods as not important meant the respondent placed more value on computer methods. Considering the "not important" rating in this way, it is clear that both groups placed the most importance on computer methods.

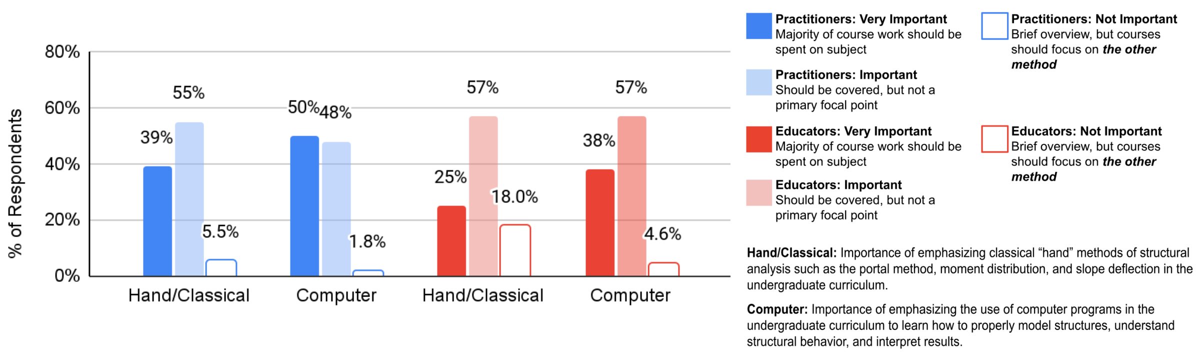


Figure 4 Practitioner versus educator opinions on importance of classical "hand" and computer methods of structural analysis in the undergraduate curriculum, [14].

The 2021 survey also included a section where participants were asked to rate the importance of 'hand' calculation methods when completing various tasks and the importance of using computer programs for various topics in the curriculum, [14]. From **Fig. 5**, computer methods were again clearly rated as critical either as a skill in itself or in applying hand methods to computer results.

Additionally, in 2021, 57% of practitioners said programing, modeling, and software use are needed to complement students' education. This is compared to only 43% responding that basic knowledge and hand calculation methods are required, [13].

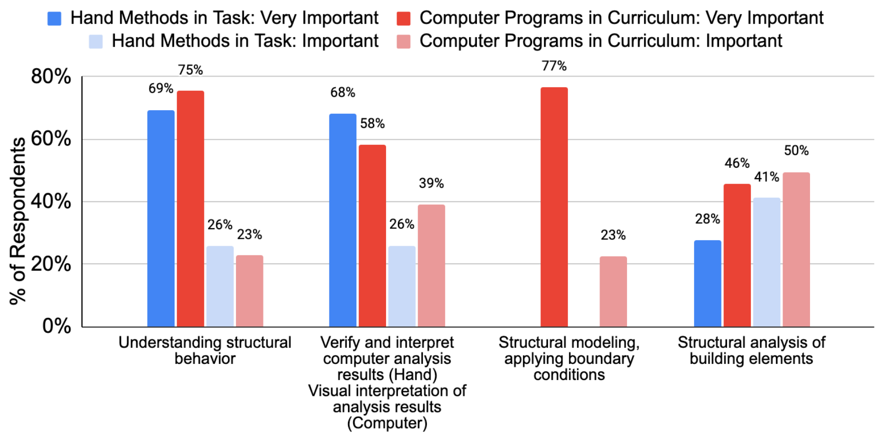


Figure 5 Practitioner opinions on use of hand or computer methods in structural analysis tasks and topics, [14] ("Structural modeling…" did not have a hand method component)

Overall Preparedness of Undergraduates

When evaluating the overall structural engineering skills of graduates, 72% of practitioners [14] and 83% of educators [15] felt that they were not well prepared to enter the workforce. Clearly there is still room to improve structural engineering and analysis undergraduate education with respect to preparing graduates for practice. Looking specifically at structural analysis courses, practitioners and educators agree that traditional classical ("hand") methods are highly important. But both groups also clearly agree (if not even more strongly) on the importance of student competency in applying the concepts, learned through classical methods, within computer programs.

From the early 1800's and the first industrial revolution [1], World War II [2] [3] [4], the computer age [6] [7], and now entering into the fourth industrial revolution with AI tools on the horizon [11], academia continues to lag behind the needs of the practice. It could be argued that some lag is appropriate and that academic training should remain rooted, mainly, in theory to provide students with the proper foundation. However, there are certainly manual calculation methods that are surely no longer needed. For example, some have pointed to the moment distribution method to develop students' ability to gain a feeling for structures, [8] and [10], but do practitioners use this method? The answer is almost certainly, no. As Powel argues in [17], carefully constructed computer software analysis exercises can achieve so much more understanding of structural behavior through experience with structures while applying fundamental understanding.

With these insights into practitioner and educator opinions, it is next pertinent to look at what educators are actually teaching. Educators seem to value computer methods, but are they being taught or used?

What is Being Taught in Structural Analysis Courses?

To get an initial sense of how current structural analysis curricula align with the NCSEA recommendations, an analysis of structural analysis course descriptions was conducted for the 268 ABET-accredited civil engineering programs in the United States. Generative AI was used to help determine common course topics in structural analysis courses. The common course topics were then used to quantify (by hand) the extent of coverage in structural analysis courses.

Methodology

The following steps outline the methodology for the course description analysis used for this study.

1. **Course Selection Criteria**
   For the purposes of this study, the "structural analysis" course at the institution was a follow-on from the strengths of materials class. In addition, the course had to be required for all civil engineering students or a breadth elective and not a course specific to one emphasis or a specialty course.[1] In cases where universities offered more than one structural analysis course[2], the course descriptions from both courses were analyzed to capture the comprehensive structural content delivered to all students. Finally, schools within the same system with identical course descriptions were treated as one course and were not counted twice in the results. Based on these criteria, 264 distinct course descriptions were analyzed for this study.
2. **Selecting Common Course Topics Using the OpenAI API**
   The 264 course descriptions were used as input to the gpt-3.5-turbo model [18] (ChatGPT) to identify common topics such as influence lines, moment distribution, etc. The first 20 course descriptions were analyzed by hand to verify that the 15 common topics identified by ChatGPT were helpful in identifying the content taught in the courses. Some refinements to the list of topics were made, including deleting topics that were represented in all course descriptions (explicitly or implicitly) and splitting others into separate categories, resulted in the following 14 topics that were used in the analysis:

   (a) Influence lines and loading (moving & live load)
   (b) Virtual Work and/or other energy methods
   (c) Displacement (stiffness) methods, including matrix methods
   (d) Slope-deflection methods
   (e) Moment Distribution
   (f) Force (flexibility) methods

[1] For example, at the Milwaukee School of Engineering (MSOE) the "Principles of Structural Engineering" was used for this study since it is a breadth elective for all students, whereas the "Analysis of Structures" course was not used because it is an elective course for the structural specialty of the BSCE.
[2]Such as Structural Analysis I and Structural Analysis II at California State Polytechnic University, Pomona

(g) “Classical” methods (or similar, e.g., “elementary methods”)
(h) Computer techniques and applications
(i) Approximate analysis methods
(j) Design concepts & methodologies
(k) Stability and determinacy
(l) Shear and moment diagrams
(m) Introduction to design loads and structural idealization
(n) Analysis of cables and/or arches

3. **Hand Analysis and Exclusion Criteria**
All 264 course descriptions were analyzed by hand to determine how many of the 14 common topics were taught in each course. The GPT-4 generative AI system was used to help verify the hand results. In compiling the final results, we excluded course descriptions that matched two or less of the 14 identified topics from the analysis because such course descriptions were typically too generic to get a good sense of the content covered.[3] This resulted in 93 course descriptions that were classified as “generic,” leaving 171 course descriptions used for the final analysis.

Results and Discussion - Course Topics

Table 1 shows the percentages of each of the 14 structural analysis topics listed from most common to least common for the 171 course descriptions used in this analysis. Comparison of these results with the survey data from practitioners reveals several interesting trends in how structural analysis is currently being taught and point to some potential misalignments with recommendations from structural engineering practitioners.

First, influence lines and moving/live load analysis was by far the most commonly mentioned topic, appearing in 64% of course descriptions. This prevalence likely stems from influence lines lending themselves well to being a distinct module within a broader structural analysis course. Determining influence lines and analyzing moving loads calls for specific techniques not required for analyzing typical static loading scenarios. Thus, many instructors appear to deliberately carve out time in their courses to cover this unique and practically relevant topic.

Another clear trend is the prominence of computer-based analysis, with 54% of courses explicitly mentioning instruction in or application of computer techniques. This aligns with the contemporary reality that computer software ubiquitously supports engineering analysis and design in practice. Civil engineering educators seem to recognize the necessity of students developing fluency with such tools.

[3] An example of a “generic” description is: “Analytical stress and deflection analysis of determinate and indeterminate structures under static and moving loads by classical methods.”

Table 1 Percentage of course descriptions mentioning each topic

Topic	% of course descriptions
Influence lines and loading (moving & live load)	64%
Computer techniques and applications	54%
Virtual Work and/or other energy methods	47%
Displacement (stiffness) methods, including matrix methods	42%
Force (flexibility) methods, (or similar, e.g., “consistent deformations")	42%
Moment distribution	33%
Introduction to design loads and structural idealization (load paths, trib. area, etc.)	33%
Slope deflection methods	25%
Shear and moment diagrams	23%
Approximate analysis methods	22%
"Classical" methods (or similar, e.g., "elementary methods", "Analytical")	18%
Design concepts & methodologies	18%
Analysis of cables and/or arches	15%
Stability and determinacy	11%

In contrast, some topic areas that practitioners highly recommend receive limited emphasis based on the course description analysis. For example, over two-thirds of surveyed practitioners rated “verifying computer results” and “understanding structural behavior” as very important skills (Fig. 4). Approximate analysis methods provide means to manually verify computer solutions, gain insights into structural behavior, and develop intuition about results. Such methods were covered in only 22% of the surveyed courses and only 11% of course descriptions mentioned **both** computer applications **and** approximate methods, representing a mismatch with their potential benefits. Additionally, fewer than one third of courses mention introducing concepts of design loads and load paths, despite over 80% of industry respondents rating load path understanding as very important for graduates. Incorporating such qualitative, conceptual aspects of analysis could better align curricula with practitioner needs.

Lastly, though not quantified here, it was observed that a substantial number of courses include lab components. Well-structured laboratory experiences present impactful opportunities to tangibly explore structural concepts like load paths and redundancy hands-on before or in tandem with computer analysis. This integration of physical and virtual experiences, informed by practitioners' needs, seems a promising direction for nurturing students’ structural intuition.

Results and Discussion - ChatGPT Analysis

Once the hand analysis was complete, we attempted to use ChatGPT to replicate the results and help check the hand analysis. The details of the ChatGPT methodology are given in the Appendix along with the Python code used for the analysis. Fig. 6 shows the number of false positives and false negatives when comparing the ChatGTP analysis to the hand analysis.[4] The hand analysis and ChatGPT matched exactly for the "Moment Distribution" and "Analysis of Cables and Arches" topics, so they are omitted from Fig. 6. Also, omitted from Fig. 6 are the "Classical Methods" and "Stability and Determinacy" topics. The "Classical Methods" item (45 false positives and 4 false negatives) was intended to match course descriptions that mentioned classic hand methods (e.g., virtual work, moment distribution, etc.) without mentioning them by name, but ChatGPT matched the topic with any course description containing the term "classical," including those that specifically identified all such methods. There were 178 false positives and no false negatives for the "Stability and Determinacy" topic. Most commonly, ChatGPT matched the term "analysis of statically determinate and indeterminate structures" to this topic while the intent was to determine the number of courses that specifically cover concepts such as the difference between unstable, determinate, and indeterminate structures and calculating the degree of indeterminacy.

The ChatGPT analysis performed surprisingly well with only 357 total mischaracterizations (false positives plus false negatives). With 14 topics and 264 course descriptions, this represents a 9.7% error rate. Neglecting the "Classical Methods" and "Stability and Determinacy" topics, the error rate drops to 3.4%. In addition, comparing the hand results and ChatGPT results identified 37 errors in the hand analysis (a 1% error rate).

Limitations

This study has several limitations that should be considered when interpreting the results:

1. This analysis focuses solely on structural analysis topics and does not examine the full civil engineering curricula at the 264 institutions. Other required courses may address skills like computing techniques and approximate methods.
2. Course descriptions provide limited information about actual course content and the depth of coverage for each topic. Analysis of syllabi and lecture materials would offer more insight.
3. The selection criteria for courses may have omitted relevant structural analysis content at some institutions. For example, elective courses were excluded though they may reinforce key concepts.
4. Benchmarking course topics does not capture the quality of instruction, learning outcomes, or how well graduates are actually prepared for practice. Surveys of recent graduates could complement this curriculum analysis.

---

[4] False positive: ChatGPT identified a topic in the course description that was not identified by the hand analysis. False negative: The hand analysis identified a topic that was not identified by ChatGPT.

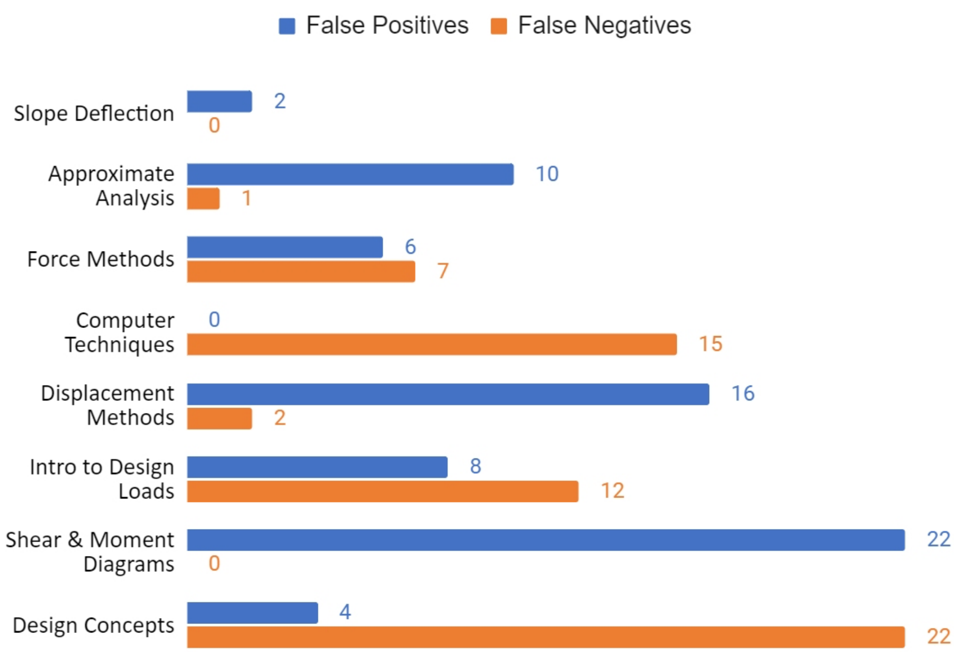


Figure 6 Frequency of mischaracterized topics when comparing ChatGPT analysis to hand analysis.

Conclusions and Future Work

A close look at the recent NCSEA surveys of practitioners and educators, [12], [13], [14], [15] revealed both practitioners and educators strongly value both classical/hand methods and computer methods in structural analysis. Each group also felt strongly that undergraduates do not emerge from their program ready to enter the structural engineering workforce. Although this sentiment was reflective of the overall structural engineering education, and not specifically structural analysis, a clear theme arose regarding classical analysis versus computer analysis methods. Along with decades of similar observations, clearly there are still improvements to be made in aligning academic training in structural analysis with the needs of the practice.

Studying the publicly available course descriptions showed that only about half of U.S. programs (~54%) explicitly claim to cover computer methods and applications. Perhaps most concerning is the lack of approximate analysis methods (~22%) in the course descriptions. These methods can be useful to verify and interpret computer analysis methods, a skill that most practitioners rated as very important (see **Fig. 5**).

The many classical methods present in the course descriptions seems to indicate that academic training is still quite focused on traditional hand calculation methods. How are these methods approached in the classroom? Are students still trudging through endless hours of hand calculations in the name of "learning structural behavior" as some believe [9] is necessary? Is that appropriate nearly a quarter into the 21st century? We suggest that classical skills need to be merged with the use of modern tools. In a similar observation, Powell [8] humorously compares moment distribution problems on licensing exams to requiring one to show they can ride a horse in order to get a driver's license. There are hard choices ahead. At some point we must give up breadth (and in some cases depth) in the interest of mastery of what is important for the profession.

As future work, the authors intend to develop a survey that probes a significant sample of educators to determine (i) what classical methods they feel are most pertinent in contemporary training, (ii) how/if they integrate computer software in their structural analysis courses, and (iii) how they plan to incorporate new (and existing) technologies in the near future. We also plan to analyze course syllabi and schedules to better understand the depth and breadth of coverage of structural analysis topics. This will give us a better understanding of topics covered and the emphasis placed on each topic.

We cannot ignore the technological tools that engineers have and must use in modern practice. Academic training is already behind in incorporating current technologies. Reconfiguring our teaching approach to incorporate the powerful analysis tools already used by practitioners is overdue. Time is of the essence to prepare for the quickly approaching AI-powered tools that will undoubtedly be adopted in the industry.

Disclosure Statement

The authors utilized large language models (LLMs), specifically systems created by Anthropic and Google, to assist in drafting portions of this manuscript and providing general language editing. The LLMs generated initial draft text for specific sections (e.g., Discussion, Limitations) and revisions at the direction of the authors and with guidance on the key points to be made. LLM systems from OpenAI were used as analysis tools in categorizing course descriptions and to help check hand analysis in parsing the descriptions. All technical content, analysis, conclusions, and opinions expressed are solely those of the human authors. The authors take full responsibility for the final paper content and acknowledge the potential for limitations in factual accuracy and potential biases inherent in language models.

Acknowledgement

The authors gratefully acknowledge the contributions of Ryder Vance, an undergraduate student at Southern Utah University. Ryder conducted the initial data collection and organization of the structural engineering course descriptions that enabled this analysis. His dedication and attention to detail were instrumental in facilitating this study.

References

[1] Aparicio, A. C., & Ruiz-Teran, A. M. (2007). Tradition and innovation in teaching structural design in civil engineering. *Journal of Professional Issues in Engineering Education and Practice*, *133*(4), 340-349.

[2] Kline, R. (1988). Reflections on the influence of World War II on electrical engineering education in the United States, 1925-1955, IEEE Antennas and Propagation Society Newsletter, 30(4), 12-16.

[3] Gregory, M. S. (1965). Teaching engineering attitude to students. *Journal of Professional Practice*, 91(1), 1-24.

[4] Gregory, M. S. (1966). Professional Training of Engineers. *Journal of Professional Activities*, 92(2), 23-29.

[5] Schiffman, R. L., & Krueger, E. R. (1972). Computer use in civil engineering. *Engineering Issues: Journal of Professional Activities*, 98(2), 251-273.

[6] Wojick, D. E. (1976). The Engineering Computer Revolution-Over or Just Beginning. *Engineering Issues: Journal of Professional Activities*, 102(3), 351-354.

[7] Wagh, V. P. (1987). Are we producing civil engineers or human robots? Journal of professional issues in engineering, 113(3), 245-248.

[8] Powell, G. H. (2008). Structural analysis: Are we relying too much on computers? Part 1: The problem. *Structure Magazine*, Nov., 50-52.

[9] Huston, E.T. (2007). Are We Relying Too Much on Computers? *Structure Magazine*, Feb., 7

[10] Criswell, M. (2004). Has The Moment Passed For Classical Solutions? Definitely Yes, Definitely No. *ASEE Annual Conference*, Salt Lake City, Utah. 10.18260/1-2--13237

[11] Solomon, E., (2021). Automation and the Future of Structural Engineering. *Structures Magazine*, Dec., 8-9

[12] Hopkins, P., Dong, K, (2018). 2016 NCSEA Structural Engineering Practitioner Survey. *Structure Magazine,* July, 34-37.

[13] Kam-Biron, M., Perkins, B., Francis, S. (2022) 2021 NCSEA Structural Engineering Practitioner Survey. *Structure Magazine*, Sept., 44-50.

[14] Dong, K., Francis, S., Perkins, B., & Kam-Biron, M. (2022, August). Practitioner and Academic Surveys + Engineering Education: A Blended Observation of Student Preparedness. *ASEE Annual Conference & Exposition.*

[15] Francis, S. M. (2021). 2019 NCSEA Structural Engineering Curriculum Survey Results. *Structure Magazine*, May, 32-33.

[16] NCSEA Basic Edu. Committee. (2019) National Council Of Structural Engineers Association Recommended Structural Engineering Curriculum https://bit.ly/3taZ0M1 (Accessed on January 21, 2024)

[17] Powell, G. H. (2008). Structural analysis: Are we relying too much on computers? Part 2: A solution. *Structure Magazine*, Dec., 20-23

[18] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei,(2020). "Language Models are Few-Shot Learners," arXiv preprint arXiv:2005.14165. [Online]. Available: https://arxiv.org/abs/2005.14165

[19] OpenAI (2023). GPT-4 Technical Report. [Online]. Available: https://arxiv.org/abs/2303.08774

APPENDIX - ChatGPT Analysis of Course Descriptions

Methodology

We adopted a pair-wise analysis approach in using ChatGPT to determine the topics covered in each course description. We scripted the API so that the LLM considered only one course description and one topic per request.[5] The LLM was asked if a given course topic was included in the course description. The LLM is necessary because of semantic differences in how course topics are described between institutions, which makes exhaustive string searches over all possible phrasings of a topic time prohibitive. If the LLM determined that a course topic was indeed covered by the course description, it returned a short (~10 word) quotation from the course description it determined was relevant. If not, it returned "no." This 10 word quotation was useful for validating the LLM's arbitrations and refining prompts with subsequent analyses. We scripted this procedure so that the output was stored in a .csv file. This calculation for 264 courses and 14 topics requires 3696 requests to a given LLM, which takes between 50 to 70 minutes (clock time, not machine time) depending on the model. We used the gpt-3.5-turbo ChatCompletion model and the gpt-4 model [19] , OpenAI's leading LLM as of this writing.

Once the hand analysis was complete, we ran new analyses using ChatGPT (both gpt-3.5 and gpt-4) and compared results. The gpt-3.5-turbo model for a given course topic would, on average, assert that some 6 *additional* topics were addressed in a course description beyond the hand analysis. The gpt-4 model offered significant improvements in completing the task, with an average error of +1 topic attributed to a given course description over the hand analysis. A comparison of the two misattributions is shown in **Fig. A-1**.

When looking specifically at each course topic, the gpt-4 model agrees much more strongly with the hand analysis than the gpt-3.5 model (**Fig. A-2**), which is encouraging for the prospect of using LLMs, albeit very capable ones, to compare curricula between institutions at the level of course descriptions. The gpt-4 model notably over-asserted that "Stability and determinacy" was contained in many more course descriptions than the hand analysis, likely because the string "determinacy" and "indeterminate" appeared in many course descriptions. This issue with this single topic is the dominant contributor to the +1 average topic error per course description for the get-4 model. While human arbitration is challenging to replace in such a task, these results show promise for implementations of LLMs to shed light on these types of questions, though we recommend only models of exceptional sophistication capable of handling logically complex tasks.

[5] The chatGPT web interface is ill adapted for an exhaustive analysis of the 264 course descriptions and 14 topics, even with the file upload function in the ChatPro interface. OpenAI's LLMs, like all interfaces, have limited context, and this constrains the complexity and size of tasks these AIs can accurately perform. The API, on the other hand, allows OpenAI's models to be scripted, e.g. with Python, leading to a more surgical implementation of their models.

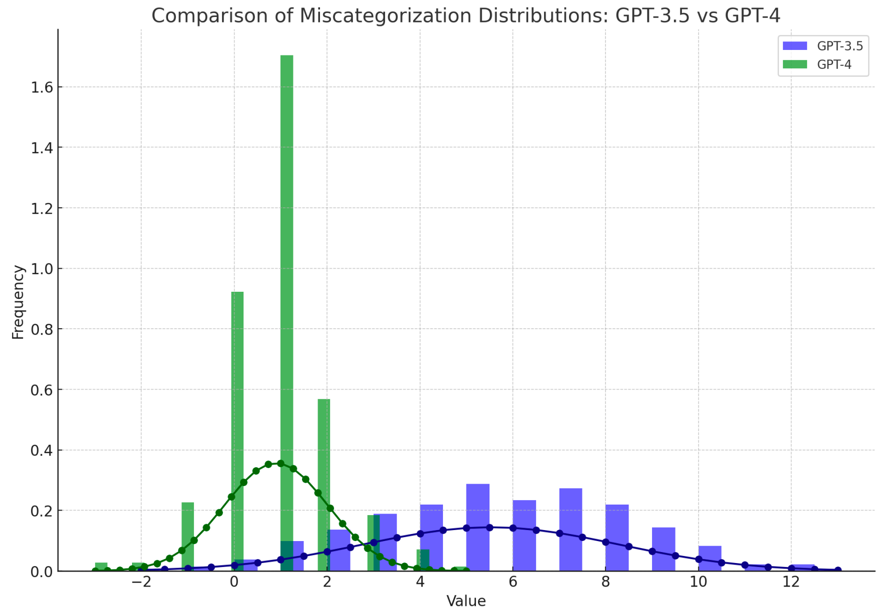


Figure A-1 Distributions of misattributions of a topic to a given course description by model.

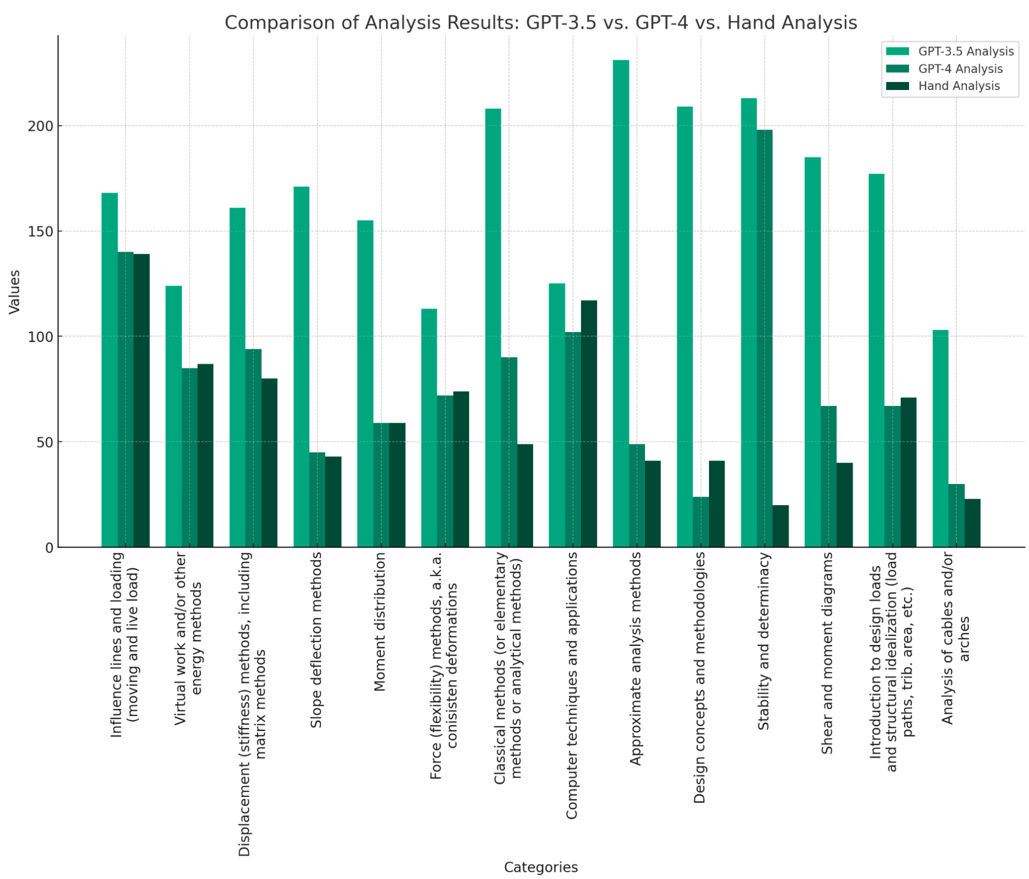


Figure A-2 Topic frequency in course descriptions as determined by GPT-3.5-turbo, GPT-4 and our manual analysis.

Source Code

```
import sys
import openai
import time
import requests
import numpy as np

# Set the OpenAI API key
openai.api_key=<API_KEY>

def chat_with_model(messages):
    response = openai.ChatCompletion.create(
        model="gpt-4”, #or “gpt-3.5-turbo”
        messages=messages
    )
    return response['choices'][0]['message']['content']

if len(sys.argv) < 2:
    print("Please provide the filename as a command-line 
argument.")
    sys.exit(1)

filename = sys.argv[1]

#
retry_count = 0
max_retries = 100 # define the maximum number of retries
retry_delay = 30

category_counts = np.zeros(15)


# Define list of course topic names
category_names = ["Influence lines and loading (moving and live 
load)",
   "Virtual work and/or other energy methods",
   "Displacement (stiffness) methods, including matrix 
methods",
   "Slope deflection methods",
   "Moment distribution",
```

```
   "Force (flexibility) methods, a.k.a. conisisten
deformations",
   "Classical methods (or elementary methods or analytical
methods)",
   "Computer techniques and applications",
   "Approximate analysis methods",
   "Design concepts and methodologies",
   "Stability and determinacy",
   "Shear and moment diagrams",
   "Introduction to design loads and structural idealization
(load paths, trib. area, etc.)",
   "Analysis of cables and/or arches"
]

#Write a csv file for output from LLM:

fp = open(f"course_description_spreadsheet.csv", "w")

#Write the first line of this csv file which will be the column
headers line of course topics:

fp.write(f'Course description \t')
fp.write('\t '.join(category_names))
fp.write('\n')


try:
   # Open the file of course descriptions
    with open(filename, 'r') as file:
       # for each course description
        for line in file:
            list_of_responses = []

            # For each category:
            for i in range(len(category_names)):
                name = category_names[i]

               # Construct prompt for AI. For each topic, check
if that topic is addressed in the course description. If it is,
return the part of the course description that addresses it.
                conversation = [
```

```
                    {"role": "system", "content": "Return up to
a 10 word response, quoting verbatim only the appropriate
portion of the course description."},
                    {"role": "user", "content": "Does the
following topic: " + name + "explicitly appear to be addressed
in this course description? If yes, quote the applicable
portion of the course description. If no, return No." + line}
                ]

               # OpenAI's API is rate-limited: an IP address
making too many requests in too short of a time will be
temporarily blocked. If this happens, we retry the request
after some period of time
                while retry_count < max_retries:
                   try:
                      # talk to this model and store the
response to the above question in response
                       response = chat_with_model(conversation)

                      list_of_responses.append(response)

                       break
                   except openai.error.RateLimitError:
                       retry_count += 1
                       print(f"Rate limit error, retrying after
{retry_delay} seconds... (attempt {retry_count})")
                       time.sleep(retry_delay)
                   except openai.error.ServiceUnavailableError:
                       retry_count += 1
                       print(f"Rate limit error, retrying after
{retry_delay} seconds... (attempt {retry_count})")
                       time.sleep(retry_delay)
                   except openai.error.APIError as e:
                       print(f"An error occurred: {e}")
                       time.sleep(retry_delay)
           # Write the course description in the first column
of the file (this is a tab-delimited csv)
            fp.write('"' + line.replace("\n", "") + '"\t')

           # write the responses of the LLM for this course
description.
```

```
            fp.write('\t '.join(map(str, list_of_responses)))

            fp.write('\n')



  # Print the list of strings
except FileNotFoundError:
    print(f"File '{filename}' not found.")
    sys.exit(1)
```