# Strategy for Integrating Design Codes in Structural Design Lectures

**Hongyan Liu**

Dr. Hongyan Liu obtained her Ph. D degree in Civil and Environmental Engineering at Colorado State University with a focus in structural engineering in 2010. Her graduate education was focused on the performance based engineering for structural systems under hazard loading, and application of innovative construction materials for underground construction. Her Ph.D. dissertation was focused on developing generalized performance-based seismic design procedures for residential buildings with a simplified format to be implemented by practice engineers. Dr. Liu is experienced in teaching structural engineering related courses and has held multiple positions as instructor at a number of engineering programs.

# Strategy for Integrating Design Codes in Structural Design Lectures

**Abstract**

Structural design classes are important contents of undergraduate civil engineering curriculum and are often required for students with a structural engineering focus. Because design codes are updated regularly, it is important to teach students the knowledge to navigate and interpret a variety of design code documents, rather than focusing on specific design equations that may become outdated when they get into practice. While fundamental concept of design principles should still be a focus of design classes, several successful strategies to incorporate the teaching of design code document into structural design lectures were proposed in this paper. A variety of active learning approaches were included to provide necessary training on code navigation and interpretation. These strategies are designed mainly for introductory design classes where students have never used codes before. A survey of past and current students were conducted to evaluate the potential impact of this teaching strategy on their professional development. It was found that majority of the students surveyed (85%) acknowledge the benefit of having code document as part of their learning experience in introductory design classes.

## INTRODUCTION

Structural design is one of the most essential components of structural engineering education. For most of civil engineering degree programs in the U.S. universities, design classes are often required for students pursuing a career in structural engineering (i.e. students with a structural focus). Currently, most of the colleges/universities offer a variety of design classes for students entering junior and senior years, including steel design, reinforced concrete design, and timber and masonry design. Sometimes, elective options for more design related topics are offered, such as bridge design, seismic design, and structural loads. While there are many class offerings related to structural design, it is also possible for a civil engineering student to complete B.S. program with only two entry level design classes. This situation makes the learning experience in these entry level design classes extremely important to students.

In a typical design class, students will perform structural analysis and design structural members based on the different code requirements. Once a student becomes familiar with load path analysis and can calculate the demands on structural members, the challenge in design is the interpretation and implementation of design codes. The ability to navigate and interpret codes is one of the most important skills students should acquire in a design class. However, for students who have always worked with instructional textbooks, the language and format of design codes are completely foreign to them. The code document is often very concise and strict, while leaving rooms for engineering interpretation in some circumstances. From a reading material standpoint, it is very different from typical textbook narratives. So it is not surprising to find that students struggle to follow codes correctly on their own without specific training. For example, the arrangement of codes sometimes requires students to go through different chapters and sections to obtain needed information. This is significantly different than a textbook where the materials needed to solve a problem are always congregated near the problem. In addition, design codes are constantly updated due to the change of technology, development of new materials, and changing practical considerations. The majority of the design codes are updated on a 3~5 year cycle. Some of the updates involve major organization changes (e.g. ACI 318 changes in the $14^{th}$ edition) and design equation changes (e.g. Chapter E7 for compression member design in AISC in the $14^{th}$ edition). Textbooks can sometimes lag behind the latest code requirements. It is a constant struggle for instructors to find an appropriate and also up-to-date textbook whenever the codes updated. Use of wrong version of textbook without adequate reference to code may result in students using outdated information in their design. While textbooks are extremely effective in teaching concepts and fundamental knowledge, here are some key skills related to the use of design code that are difficult to teach using only textbooks.

In the existing literature, educators have pointed out the importance and unique characteristics of structural design education. Aparicio and Ruiz-Teran [1] provided a historical perspective on design education, with a central question on what should be taught and how to most effectively deliver. Over the past decades, educators have looked at different ways to teach structural design, including problem and project based approaches (e.g. Quinn and Albano [2], Mills and Treagust [3]), as well as teaching tools utilizing web-based technology [4][5]. While code document is an inseparable part of design, there has not been dedicated discussion in existing literature (based on the author's limited experience) on the adoption strategy for code documents in lectures. In this paper, a viable approach to integrate design codes into introductory design class instruction was summarized. In the following section, the strategy for developing the students' skill set related to code document utilization is outlined. Representative examples for implementation of this strategy are described using different code subjects. Finally, survey results on use of code related knowledge from students at different career levels are presented. It is illustrated that effective incorporation of design code documents into structural design classes will be beneficial to students in their professional career.

## IMPLEMENTATION STRATEGY

Because of the difference in writing and organization style between design codes and textbooks on structural design, using the textbooks as a sole reference in design classes may not fully prepare the students for the skills they need in their future profession to effectively navigate and interpret codes. Since there is no required classes in typical U.S. higher education curriculums to mandate the full introduction of code document, it is possible for BS graduates to not have a full understanding of the code system. Therefore, it is recommended that following strategies of implement codes into design classes be considered by structural engineering educators in required entry level design classes.

1. Make code a required material for taking the class

The first step of implementing codes in design classes is to require students to own a copy of the design code, either physically or electronically. While this step seems trivial, it is not universally practiced by all instructors for a number of reasons. Some instructors may view this as an unnecessary cost (i.e. burden) to students because it is impossible for any 3-credit entry level design class to cover most of the code provisions. Some instructor opt to use hand-outs that contains part of the code, or simply rely on the textbook used. However, the investment on full code document is a worthy one for students who have interest in design career, even beyond the class itself. Nowadays, most of the professional institutes who publish the design codes, such as American Institute of Steel Construction (AISC), American Concrete Institution (ACI) and American Wood Council (AWC) etc., offer some form of discount for students and instructors on the code product. Some of these discount was attached to the student membership which is free for all the in major college students.

In addition to the discount program offered by publishers, there are several additional approaches an instructor can try to help students with code access. For example, connections can be established by the instructor with local industry to provide sponsorship for design code purchase for students (this was done at the author's institution to provide free NDS (National Design Specification for Wood Construction) package for students taking wood design). In addition, some code publishers offer free online access to certain codes (e.g. IBC, NDS). The instructor can help students financially by allowing and accommodating the use of online codes. Because most of the codes do not change significantly over versions, the instructor can allow use of earlier versions (likely much more affordable if buy used) by providing supplement handouts.

2. Dedicate class time to teach code structure and navigation

The second, which is the most important step of implementing codes will be teach students techniques to navigate the code. Specific formula and requirements may constantly change but the principles of organization and navigation within code document is consistent. This can start by spending one lecture on the overall structure of the code and specifically identify the provisions that will be used for this class. Instructor need to create a friendly environments to introduce design codes by engaging students with hands-on activities to browse through code. With limited lecturing time, it is not possible to cover all details within the code, but its structure and organization can be taught without knowing all details. After the main structure was introduced, the lecture can be focused on a specific design topic, while guiding the students through the steps they should follow. Specifically, the instructor could pose questions to students on details of the code provision that will lead them into references and commentary documents. In other words, it is less important for the instructor to let students know or remember specific code formula or content, but to cultivate the students' ability to self-navigate the code.

Another important aspect of teaching with code is to set an example for code language interpretation. The instructor should use suitable examples and code provision to illustrate that there may be different interpretations on code requirements that would lead to different designs that are all acceptable. The notion of having more than one "standard" solution will be quite foreign to undergraduate students at this point (if this is their first design class). Such an experience is not commonly available through the use of a textbook, where most of the problems have standard solutions.

3. Introduce code updating process and highlight code changes

While it is common knowledge to design professionals, entry level design students are likely unfamiliar with code updating process. It is important for instructor to demonstrate to students the potential changes in the codes using real-life examples. Again, then emphasis should not be on remembering these specific changes, but to imprint in students' minds that design code is a living document that will constantly change during their career. A brief introduction on the process of design codes adoption into IBC and local jurisdiction will also benefit students greatly.

4. Assign class practices focusing on using code

Use of code document during in-class active learning is a highly efficient instruction technique that will benefit students greatly. After explaining the structure and navigation of the code and illustrated these key techniques using examples, it is critical to provide an opportunity for students to practice using the code themselves. This activity is better done in class with encouragement for discussion and teamwork. The practice problems do not need to be comprehensive to start with. Even activities like looking for cross section properties in AISC manual will work. It is important to apply common active learning techniques to encourage participation, such as calling on individual student for quick checks to keep them engaged.

5. Use bonus credit to motivate self-exploration beyond instructed material

Finally, in order to test if students truly independently mastered the use of design codes, it is a good idea to single out certain problems or topics for students' self-study. These topics should not be covered in the instruction but be relatively easy to solve if students know how to navigate the code for applicable provisions. Because this activity requires higher level of understanding, it is recommended that these activities be assigned outside of the mandatary requirements for the class. Careful use of bonus credit activities (in homework or exams) will be very effective to encourage self-guided exploration of the code material for motivated students.

## TEACHING PRACTICE EXAMPLES

In order to illustrate specific measures instructors can take to implement the strategy outlined above, a few selected examples are presented in this section.

### Example 1: Introducing Code Structure/Organization

At the beginning of the class, it is recommended that the instructor dedicate up to one lecture to introduce the main structure of the code document, and lead students explore the entire design code.

The process of introducing AISC Steel Construction Manual, 15$^{th}$ edition [6] (will use AISC Manual hereafter for abbreviation), is used here as an example to demonstrate process. The instruction should first focus on the big picture structure of the code document before diving into the part of the provisions that will be immediately used. Hands-on activities can be implemented to get students familiar with the code organization, such as an in-class competition on standard cross section property look up in groups. Once the students get familiar with one chapter (e.g. tension design chapter), it is a good practice to zoom back again at the big structure and go over different chapters in the steel code (briefly) again to reinforce their recognition of the code structure. This combination of alternation between detailed provision instruction and general structure explanation will help establish the general concept that the code is a tool with inter-connected parts. For example, the majority of the steel manual (Part 3 to Part 15) is simply design tables and charts developed based on these individual code chapters, instead of some black magic designers have to follow. The difference between code provisions and commentary should also be highlighted on the conceptual level, as well as the connection between the Manual requirements and ASCE7 (Minimum Design Loads and Associated Criteria for Buildings and Other Structures). This could naturally lead to an introduction to the larger building code environment including IBC. It is generally a good practice to incorporate the introduction of the code to students during early phase of the semester, right after basic structural materials are introduced. The introduction should be done in one lecture and cover the entire scope of code document.

Slight alteration of this technique can be applied to other codes based on their unique structure. Wood design package consists of separate booklets such as the main NDS design manual and the lateral design provisions (SPDWS). NDS was organized following different engineered wood products, thus it is beneficial to delay the introduction of code until basic introduction on engineered wood products is covered. Concrete code (ACI318) [7] has gone through major re-organization from a behavior-driven structure to a component based structure similar to that of the steel code.

**Example 2: Hands-On Practice on Code Navigation**

Unlike text books or scientific articles in which the authors always try to organize material and narrative around one or a few focused subjects, design codes are more similar to legal documents with requirements and cross-references scattered around. Teaching students the ability to navigate the code provisions to find relevant details and requirements related to their design problem is very important if we expect students to be able to solve real life design problems (vs. a simple problem in the textbook). It is best to use guided hands-on active learning activities to convey this skill. Here we will use two examples to illustrate active use of code in class to engage student learning.

In the first example using AISC Manual, we consider an in-class practice to have the students find the net tension cross section area of a $W\ 12 \times 96$ section with one drilled standard hole in the web for a 7/8" bolt. This is a great example because while the problem is simple, the students need to go through a number of different chapters in the design code and also some reference tables to truly understand the organization of the code. The following steps should be followed by students using their own Manual with the help of the instructor:

1) Go to *Part 1 Dimensions and Properties*, find Table 1-1 for W-shapes to obtain basic section properties.
2) Go to *Part 16 Specifications and Codes Chapter D Design of Members for Tension* to find how to calculate effective net area $A_e$ in *section D3*, here the students will be directed by the code provision to *Chapter B Design Requirements*

3) Go to *Chapter B Section 4.3b*, have the student read the definition of net area $A_n$, note the requirement to add $\frac{1}{16}$" $(2\ mm)$ to drilled hole diameter.
4) Explicitly point out to students how the difference between the terms "bolt diameter" and "hole diameter" is presented in the code. Then proceed to *Chapter J Design of Connections* to introduce different types of holes
5) Go to Table J 3.3 to do a hands-on activity to look up standard hole diameter for 7/8" bolt, while reminding them the code provisions covered in steps 2-4.
6) Put everything together and teach students to calculate net area $A_n$ following *Chapter B4.3b*.

While listing all correct formula and steps, many textbooks will not explicitly go over the location of the formula during this process in the code manual. The idea is to teach students this full "pathway" for net area calculation following the code. Teaching the full path will prepare students for future code changes and updates in case some details are changed. Instructors can choose to go over steps 1-6 with any preferred instruction tools they like, such as class discussion, or guided note-taking. Students' understanding of AISC code will be greatly enhanced this way instead of simply give them the correct answer.

Similar example can be developed for teaching Building Code Requirements for Structural Concrete (ACI318-14). For example, a useful exercise of finding minimum thickness of a two-way slab can be used as a gateway to learn how to navigate ACI code. In this example: the slab is not prestressed, it is an interior panel without drop panel, GR60 rebar($f_y = 60\ ksi$) should be used. The following steps may be followed by students using their own ACI318-14 under instructor guidance:

1) Go to Table of Contents, use key words "two-way slab" to find *Chapter 8 Two-way Slab* under "Members" category. Instructor can also throw a discussion thread for students to discuss the difference between one-way slab and two-way slab.
2) Read the list of the sections under *Chapter 8*, here the instructor can give an overall review for each section then let students decide which section should be used for this problem. The correct answer will be *Chapter 8 Section 3 Design Limits*.
3) Go to Section 8.3 Design Limits, instructor will give students some time to read the entire section. Then students should be able to find the code table for minimum thickness (*Table 8.3.1.1 Minimum thickness of nonpressed two-way slabs without interior beams (in.)*).
4) Instructor can organize a discussion with students on the use of Table 8.3.1.1. For example, the instructor should use realistic photos from concrete building projects to visualize what a drop panel is. Through the discussion, students will learn how to pick the correct design limits/requirements based on their own projects and design codes.
5) Instructor can use this opportunity to illustrate the importance of reading the code provision details. For example, the definition of $l_n$ in Table 8.3.1.1.1 is "the clear span in the long direction, measured face-to-face of supports (in.)". Instructor can list all the key words that will affect design calculation, such as "clear", "long direction", and "face-to-

face measurement”. Instructor can further point students to *Chapter 2 Notation and Terminology* to look up different terminologies and their differences.

**Example 3: Bonus Material to Encourage Self-Exploration**

Due to limited lecture time during the semester, it is not possible to cover all materials within any design code. This provides an opportunity to incorporate self-study into student learning through entry level topics. Even in their first entry level design class, if basic understanding of code is established, students should be capable of using code independently to solve simple problems not covered directly in the lecture. Assignment related to these self-study tasks should be given with enough free time for students to study and explore, such as an extra-credit homework problem. It is critical that the instructor make it clear that there is no penalty for students to try these bonus problems under any circumstances. While not negatively affecting their grades, students' performance in these self-study activities can be used as an objective assessment on students' ability to independently navigate and apply the codes.

1. Steel design: Design of eye-bar members in tension member design. This subject is covered in Chapter D of the code as an independent section. While it has some special requirements, the concept is very straight forward and similar to other tensile member design that is typically covered in lectures.
2. Concrete design: Reinforcement detailing in cast-in-place wall design. While being an advanced topic, if the loads added are only compression and out of plane bending, the behavior of the wall can be considered as one-way slab, hence the students should have the needed knowledge to do it. The new contents might be the layout of the reinforcement for walls, which can test the students' ability to self-explore and identify those requirements.
3. Wood design: Compression design for round pole members. The concept and procedure for round pole design is the same as compression design for glulam or lumber members, with the difference mainly lies in the reference values and adjustment factors. It is listed separately as a different chapter, which provides the students with the experience of self-exploration of a new engineered wood product. Design of round pole member is typically not covered at entry level wood design classes (especially if the lecture is a wood-masonry combined class) but fairly important in practice.

## STUDENT SURVEY REGARDING STRUCTURAL DESIGN AND USE OF CODE

While the proposed teaching strategy about codes was developed based on the author's teaching experience, it is useful to also investigate the opinion of students in order to validate the need for suggested strategy and methods regarding code incorporation in classes. A survey is designed to reach out to current and former students who had taken design classes. A total of 26 students responded to the survey, including 8 current students and 15 students already joined the work force. These students are used to represent typical structural-focused student body that satisfied following criteria (1) all students have experience taking at least one design class while they were in school; (2) all of the surveyed students are either working for a structural design firm or have strong interest to do so after graduation.

The survey was designed to gather two types of information: 1) how do these students use design codes in their work (job or school work); and 2) how do they value the knowledge of code in their education. Firstly, the students were asked about the percentage of work they are conducting now that require “structural design knowledge” and “active use of Codes”. The resulted histogram from the responses is shown in Figure 1. The result indicates that about half of the work a structural-focused student do requires the use of different codes. This confirmed the need and benefit for including codes in our classes. There are even more responses indicating the importance of design knowledge, which includes the ability of code-reading and implementation discussed before.

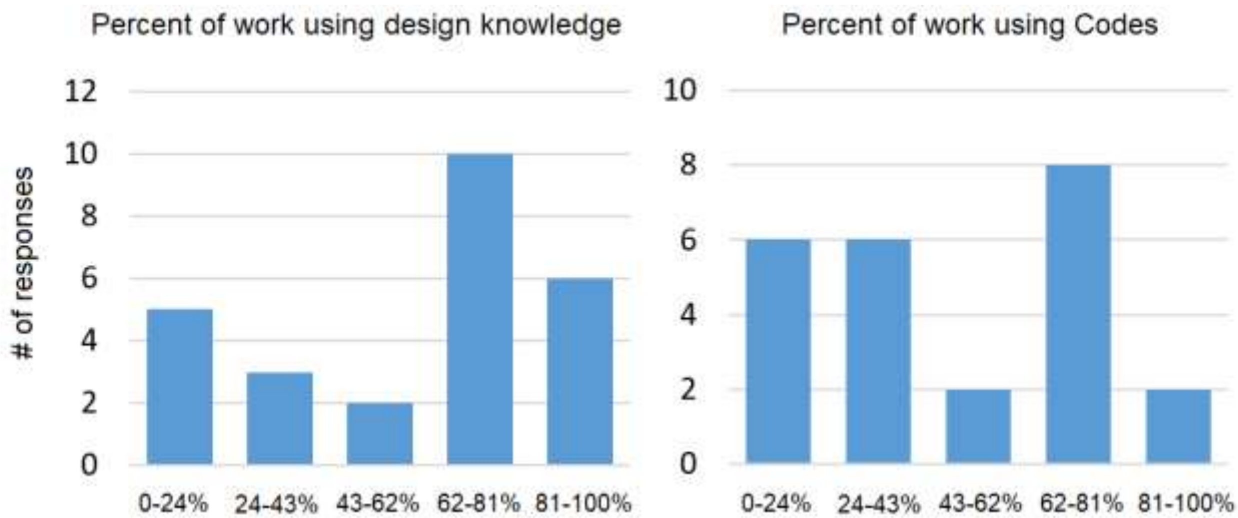


Figure 1: Histograms for design and code usage of surveyed students

The next question asked about specific code used. There are 5 levels of usage frequencies defined in the survey (0-Not at all, 1-Rare, 2-Not that often, 3-Often, 4-All the time), which are used to calculate a usage frequency index for different codes within the surveyed group. This index was calculated by multiplying the usage frequencies number with the number of people fill in that index and adding all survey responses together. The resulted usage frequency index is then normalized (divided by the number of responses, so the maximum possible frequency index is 4.0) and shown in Figure 2. Certainly the use of different codes will depend on the type of work or firm the students are involved in. For the limited amount of people surveyed, ASCE7 is most often used, followed by the steel and concrete code, which matches the common practice in design. This information can be valuable to instructors of different design codes to prioritize their efforts on introduction of different codes. For example, it is recommended that there should always be an effort to introduce ASCE7 in any structural engineering program.

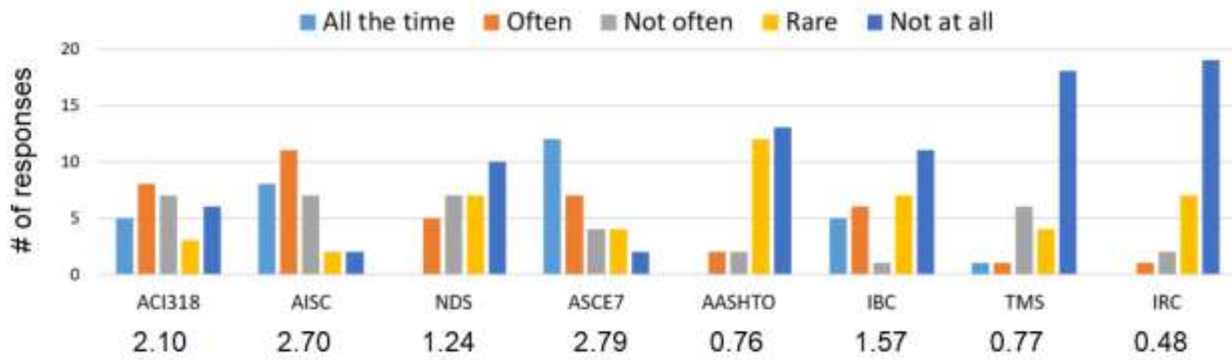


Figure 2: Bar chart and usage frequency index for different codes

The rest of the survey focused on the students' experience and opinion on the use of design code in classes. As it can be seen in Figure 3, most of the surveyed students obtained their first experience of the full code (read and understand the overall structure of the entire code document) in entry level design classes. While this result is biased toward students from the author's program, it is interesting to see that familiarity to the full structure of some codes may not be necessary even for design professionals surveyed in this study. It can be seen the structure of AISC code is relatively easy to teach for the students surveyed.

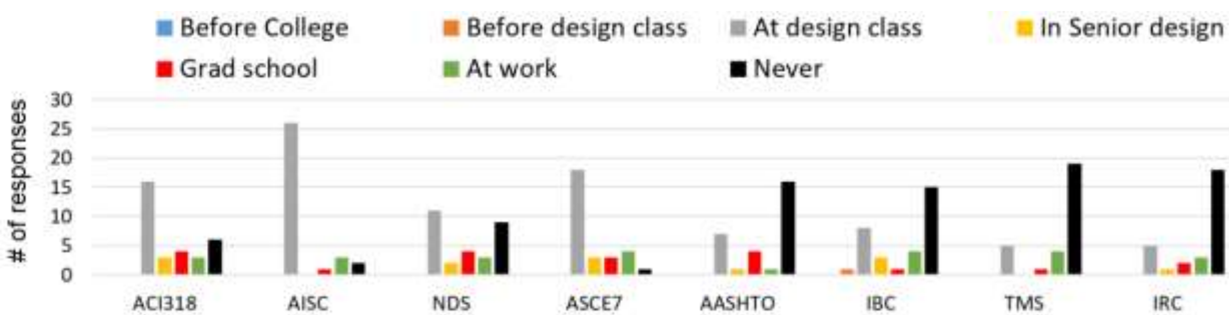


Figure 3: The timing of full inspection of the code document

The following question was presented to get the students' opinion on the value of purchasing a copy of full code (compared to relying only on textbook) for design classes. Vast majority of the surveyed students (86%) felt it is worth it, which partially support the proposed method of code material inclusion in classes.

Finally, a survey on the importance of knowledge points in design class instruction was conducted. The students were given a discrete grade of 1~7 to grade each topic with 7 being the most needed (i.e. Essential). The responses were aggregated and then divided by the total number of responses, thus returning the weighted average importance index (also ranged in 1~7) for each topic. The results are shown in Table 1. The most important knowledge points (where to find equations and knowing the code structure) identified by the students agree with the main hypothesis of this study, that explicit teaching of the code document as a whole is important, as well as the focus on developing students' ability to navigate the code.

Table 1: Summary of importance of knowledge points

| Knowledge point | Average Importance Index |
|---|---|
| Remember specific design formula | 2.86 |
| Know where to find certain design equations in Code | 6.07 |
| Know the structure of the Code | 5.66 |
| Be able to solve examples and homework problems in textbook | 5.03 |
| Understand the reason behind formula in Code and how to derive them | 4.34 |
| Learn how to read and understand Commentary of the Code | 5.55 |
| Discuss and compare different version of the Code | 3.41 |

Similarly, another survey question has the students use the same rating system on the helpfulness of different learning activities typically employed in a design class. The results from this question are listed in Table 2. It is apparent that students saw value in collaborative work, which is reflective of their design work environment. This data indicates that in-class hands on practice is one of the most productive ways to enhance learning outcome, which matches several suggested implementation methods using codes outlined in this study.

Table 2: Helpfulness of different learning activities

| Learning activities | Average Importance Index |
|---|---|
| Work on HWKs individually | 3.62 |
| Work on HWKs in a study group | 4.48 |
| In class hands on practices allowing discussion | 4.76 |
| In class hands on practices done individually (small quiz) | 2.48 |
| Comprehensive group projects | 4.48 |
| Projects done individually | 3.38 |

In addition, there are two free-response questions to obtain suggestions regarding how to teach structural design and use design code. The key responses highlighted the need to more comprehensive project based problems for design, direct use of code and commentary, and the introduction to model codes such as IBC and ASCE7. Some of these recommendations can be most adequately addressed by devoting a lecture or two on these related code documents in an entry level design class, while others will likely have to be incorporated in a final senior design type class instead of entry level material code design class.

## CONCLUSIONS AND RECOMMENDATION

A strategy to better incorporate design codes in structural classes was proposed in this study with implementable methods and examples. This strategy was designed to better prepare structural engineering students for their future career if their tasks involves structural design. There are a number of active learning activities that can be incorporated in the teaching of code references. A small scale survey on current and past students in structural engineering showed that students value the experience of learning the actual code document in school, and the most important aspect of code design instruction is to provide opportunity to understand the structure of the code in order to navigate it for needed formula. Students also values the experience of applying code and design approach to a complete system rather than at only the component level. However, such experience will typically be attached to higher level structural classes (e.g. senior design) instead of entry level design classes due to time constraints. Based on the strategy and data presented in this study, it is recommended that design codes should be introduced to students in its entirety for structural design classes and be used to promote active learning for students.

Finally, for new faculty without industry design experience or even senior faculty who has not been teaching design for a while, special attention needs to be paid in preparation of design classes when adoption the proposed code-focused strategy. It is recommended that instructor spend some time before the semester to get familiar with current code versions and re-work old examples if needed. Getting connected with local industry is always a great idea for faculty at any level. Based on the author's experience, the local firms are typically excited about getting involved in design education as they will potentially hire form the class. The format of collaboration can span from giving guest lecture on specific design topics to provide practical design examples for students to study.

## REFERENCES

[1] Aparicio, Angel C., and Ana M. Ruiz-Teran. "Tradition and innovation in teaching structural design in civil engineering." Journal of Professional Issues in Engineering Education and Practice 133, no. 4 (2007): 340-349.

[2] Quinn, Kerry A., and Leonard D. Albano. "Problem-based learning in structural engineering education." Journal of Professional Issues in Engineering Education and Practice 134, no. 4 (2008): 329-334.

[3] Mills, Julie E., and David F. Treagust. "Engineering education—Is problem-based or project-based learning the answer." Australasian journal of engineering education 3, no. 2 (2003): 2-16.

[4] Adeli, Hojjat, and Hongjin Kim. "Web-based interactive courseware for structural steel design using Java." Computer-Aided Civil and Infrastructure Engineering 15, no. 2 (2000): 158-166.

[5] Elhouar, Souhail. "Teaching steel design on the web: The Blackboard experience." In American Society for Engineering Education, pp. 1-2. 2005.

[6] AISC, (2017) AISC Steel Construction Manual, 15th edition, American Institute of Steel Construction, Chicago, IL.

[7] ACI, (2014) Building Code Requirements for Structural Concrete (ACI 318-14) and Commentary (ACI318R-14), American Concrete Institute, Farmington Hill, MI.