AC 2012-2960: EXERCISES FOR STUDENTS TO LEARN THE PROPER APPLICATION OF ANALYTICAL COMMERCIAL ENGINEERING SOFTWARE

Dr. Aaron P. Wemhoff, Villanova University

Aaron Wemhoff earned a Ph.D. in mechanical engineering in 2004 from the University of California, Berkeley. He worked as a Thermal-fluids Engineering Analyst at Lawrence Livermore National Laboratory prior to beginning employment as an Assistant Professor of mechanical engineering at Villanova University in 2008.

Exercises for Students to Learn the Proper Application of Analytical Commercial Engineering Software

Abstract

A series of self-learning exercises instructs students in the proper application of engineering software that includes choosing optimal input parameters, validating the results with theory, and determining the appropriate mesh resolution. These exercises also enable students to gain confidence in their solutions by supplementing the computational results with theoretical and/or experimental comparisons. The feedback from the students indicate resistance to the self-learning exercises compared to a traditional learning model, although the majority of students realized the importance of validation and gained ideas into approaching the learning of new software.

I. Introduction

Professional engineers are often tasked with solving difficult problems. Correct solutions to some of these problems are critical to personal safety, national security, and a manufacturer's economic viability. Incorrect solutions can lead to poor engineering design, which has been documented in several places to be the cause of physical harm or death[1-3]. It is because of these potential catastrophic results that engineering disasters are regularly studied in engineering ethics courses[4-6].

The reliability of the solutions is related to the confidence the engineer has in the solution method. This connection has been made formally by defining a solution's confidence level by examining the uncertainty in design parameters[7], but this relationship is difficult to formulate for an academic setting. The confidence in the solution method for a student can be built by following a worked-out example, matching the final answers with the answer key, or applying multiple approaches to tackling a problem. Lecture courses often use the first two approaches due to the simplicity of problems and the availability of example problems in textbooks. However, in professional practice the problems have not been previously worked out and solutions are not given. Therefore, the only means to gain confidence in a solution by a professional engineer is to apply multiple approaches to tackling a problem, and improved reliability is achieved when the two different routes to a solution give similar results.

Figure 1 shows the advantages and disadvantages of applying the three problem-solving approaches: experimental, theoretical, and computational. Experimental approaches can be too expensive or require too much lead time, and in some cases the use of experiments is not allowed due to legal or political reasons[8]. Theoretical approaches may oversimplify problems by applying invalid approximations and assumptions to make it analytically solvable. Likewise, computational approaches may inherently contain simplifications that may not reflect reality. Furthermore, computational approaches can often provide solutions that theoretical approaches cannot feasibly solve, but the risk of a poor computational setup will result in poor answers. For this reason, software users must recognize these limitations and perform validation of their models to ensure the reliability of results[9].
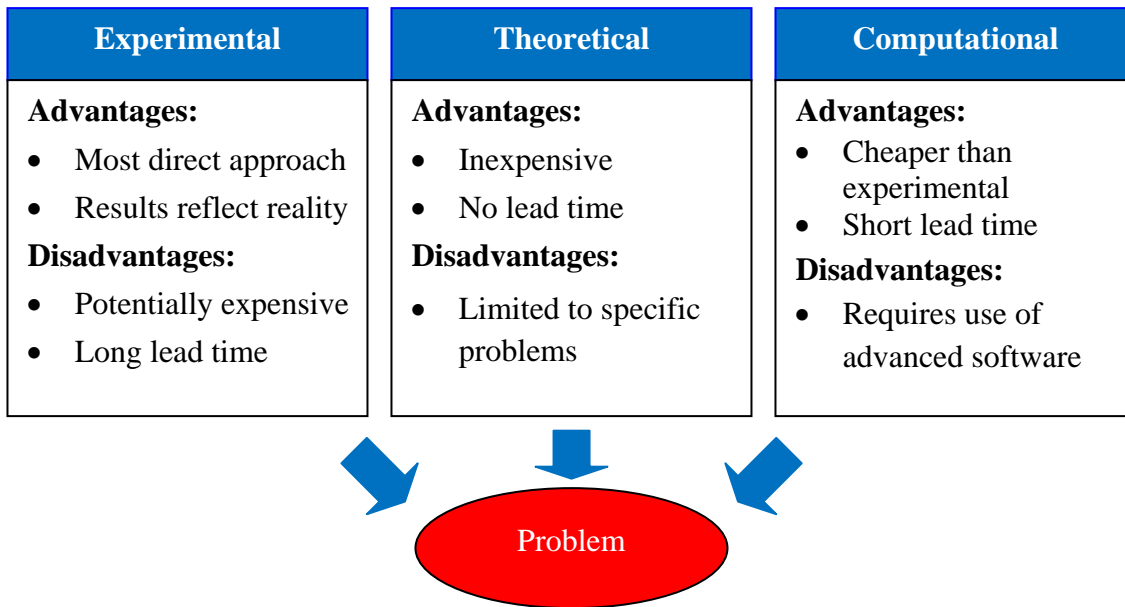
Experimental	Theoretical	Computational
Advantages: • Most direct approach • Results reflect reality **Disadvantages:** • Potentially expensive • Long lead time	**Advantages:** • Inexpensive • No lead time **Disadvantages:** • Limited to specific problems	**Advantages:** • Cheaper than experimental • Short lead time **Disadvantages:** • Requires use of advanced software

Problem

Figure 1. The advantages and disadvantages of individual approaches to problem solving, thus requiring the use of multiple approaches towards problem solving to gain confidence.

The purpose of this study is to emphasize the use of multiple approaches to problem solving as applicable to professional engineering practice. The limited time and resources available for the study caused the study to focus on combining theoretical and computational approaches rather than incorporating experimental approaches, although the use of experiments to compliment these other approaches is valid in general. To illustrate the aforementioned concepts, the end goal of the study is to have the students use two approaches (computational and theoretical) to gain confidence in solving a problem with no available theoretical solution. This goal is achieved by performing validation of new computational software. The gained confidence is then translated into applying the code in a reliable manner to achieve new results – in this case, the Nu correlation for natural convection surrounding a nonstandard geometry. The remaining portions of this paper discuss the method used to achieve these goals and the results of the study.

II. Method

The study began with the choice of software package that is applicable towards Mechanical Engineering. Professional engineers generally use three types of software to handle their analyses of various problems: Computer-Aided Design (CAD), Finite Element Analysis (FEA), and Comptuational Fluid Dynamics (CFD). Many Mechanical Engineering students are only exposed to CAD and FEA in the core curriculum. Furthermore, students are taught theoretical Fluid Mechanics concepts in the core curriculum, so the use of fluids-based problems with CFD software is a natural fit for this study. Therefore, in the study fluid mechanics problems were solved using a computational approach (CFD) while the software is validated using a theoretical approach. The educational software EasyCFD[10] was used in the study for the following reasons:
- EasyCFD is much less expensive than the more popular commercial codes Fluent[11], Star-CD[12], and the open source code OpenFoam[13].
- EasyCFD contains fewer features than Fluent, Star-CD, and OpenFoam. The lack of features is advantageous for training undergraduate students on CFD in that they will be

less likely to be overwhelmed by the features when exploring the software. For example, the fact that EasyCFD only performs 2-d simulations makes mesh generation easier compared to the other 3-d codes.

- EasyCFD is relatively small and therefore quick to install on standard laptops.

The general approach for this study is to develop assignments that involve student self-teaching of the CFD software. This enables students to "learn how to learn" new software. The study was applied as part of the Thermal-Fluid System Design course, a junior Mechanical Engineering course provided to those students choosing to concentrate their studies in the thermal-fluids arena. These students applied the software to work on five exercises throughout the Spring 2011 semester. These exercises contained minimal guidance on how to get the achieved result. At the end of the semester the students were provided with the instructor's results to allow the students to go back through the exercises.

The exercises were developed based on the anticipated experiences that the students will likely face as professional engineers: they will be presented with a problem that will require the use of new tools, whether it is analytical relations, experimental apparatuses, or computational software. In this study, the development of confidence in the use of a new software package should follow a set of guidelines that enables the user to confidently apply the new tool towards the problem of interest. In order to achieve this goal, the exercises outline how the user should do the following procedure:

1. Follow a tutorial provided by the software manufacturer to gain experience in apply the general software features.
2. Apply the software towards simple problems to validate the use of all pertinent code features to the problem of interest.
3. Extend the modeled system to the problem of interest.

In the case of the following set of exercises, the end goal is the development of a Nusselt relation for natural convection surrounding a nonstandard geometry. This application of CFD acts as a supplement to the fundamentals of natural convection that are taught as part of the standard curriculum in the Thermal-Fluid System Design course.

The five exercises are as follows:

1. **Measure the drag on a cylinder in air.** The exercise showed the students that following a tutorial is the easiest means to get comfortable with new software. Furthermore, even when following the tutorial the students see that certain aspects of the simulated results (i.e. drag force) do not agree with experimental results.
2. **Drag on a cylinder: find the best input parameters.** The exercise showed that adjusting the input parameters influences the output parameters, and it demonstrated that the best combination of parameters for one output may be detrimental to other outputs. It also showed that confidence in using the software may be gained from matching experimental results.
3. **Drag on a cylinder: mesh resolution study.** The exercise showed that the mesh resolution has an impact on the calculated outputs. The exercise also provided the students practice with determining the best mesh resolution.
4. **Entrance region flow in a pipe.** The exercise allowed the students to practice setting up simulation input parameters such that desired output parameters can be compared to

known values. It also provided a means to validate simulations by comparing the simulation results to the known solutions.

5. **Natural convection heat transfer from a nonstandard geometry.** In this exercise, the students learn to validate the new code first by performing the analysis on a standard geometry with a known result, which allows them to get a sense of the error in the calculations. Then, simulations on the nonstandard geometry were performed with the anticipated error in mind.

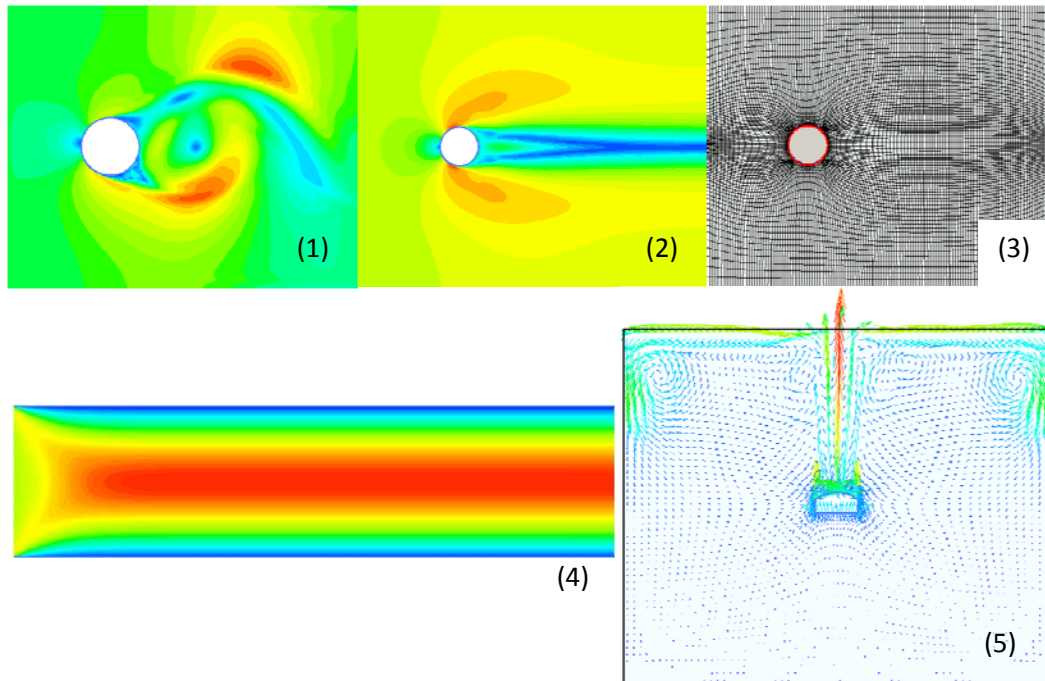Screen shots from EasyCFD for each of these five exercises are shown in Fig. 2 below.

Figure 2. EasyCFD screen snapshots for each of the five exercises in this study.

The choice of the end goal for the set of exercises in this study is the development of a Nusselt correlation for a nonstandard geometry to supplement the natural convection theory discussed as part of the course. The result, shown in Fig. 3, connects the use of CFD with theoretical correlations discussed in the course lectures. The figure shows that the error in Nu values is rather large from the comparison of calculated and theoretical Nu correlations for a infinite cylinder, yet the data in the figure do appear to be well-fit by the power-law trendline shown.
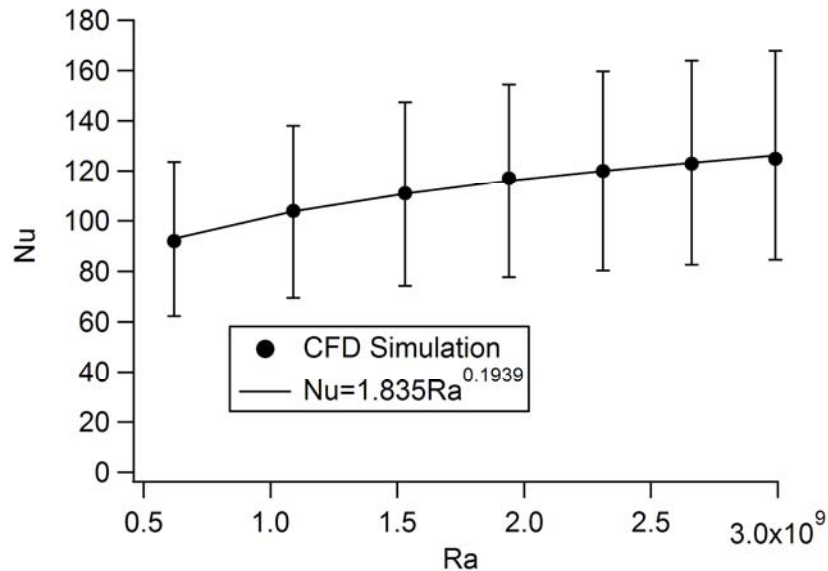
Figure 3. The Nu correlation developed using the CFD exercises for natural convection surrounding a nonstandard geometry.

It should be noted that one aspect of the exercises is to train students to find which data are important for validation purposes. To direct the students to the important data, they are instructed to locate and provide values for specific parameters. For example, in the second exercise they are instructed to indicate the difference in calculated total drag force on the cylinder, which requires the ability to obtain the values of drag force from the simulations. In addition, it should be stated that the above exercises could be tailored to other courses, for several 2-d CFD validation cases are readily available online[14].

It should also be noted that these exercises focus on validation ("am I solving the right equations?") as opposed to verification ("am I solving the equations right?"). From the software applications viewpoint, the end users are more concerned with validation whereas the code developers provide the appropriate verification. However, it could be argued that knowledge of the software can provide verification for the implementation of the theoretical correlations that are used for comparison with the results of the software. For example, the third exercise could be considered from the viewpoint as providing a verification exercise for determining the theoretical entry length of a developing flow.

III. Results

The initial observations from the study showed that many students had difficulty in self-teaching since they spent a great deal of time on the exercises, which eventually caused them to get frustrated. The class could be broken down into 3 groups: 13 students attempted and understood most of the exercises, 4 students attempted some (but not all) exercises, and 5 students attempted little or none of the exercises. 17 of the 22 students took a survey at the end of the semester to provide the feedback shown in Table 1. The survey shows that although the majority of students learned new ideas as to how to approach learning new software (Question 8), they did not experience major changes in their overall approach (Question 9). The difficulties faced by the students in performing the exercises are seen in high values for Questions 1 and 4. Note that

Question 5 is related to the problems in installing software using a license server, which provides a lesson for the instructor in that effort is required to provide the students the appropriate software licenses.

The exercises were designed to achieve the goal of giving students confidence in applying new software towards designing the Nusselt relation for natural convection around the nonstandard geometry in Fig. 5a. In retrospect the following changes will be considered for future implementation of the methodology presented here:

- At the initiation of the exercises, the students would be given a clear pathway for the process of learning new software. Therefore, they would see how all exercises fit into the general approach of confidently applying new software towards an advanced problem.
- The second exercise would involve the application of a mesh resolution study to show the importance of grid refinement in achieving the correct model.
- The third exercise would involve the variation of input parameters to see how they affect the result. The students would then use this variation to obtain an idea of the error in the computed values.
- The fourth exercise would be replaced with a natural convection model of a standard geometry (e.g., an infinite circular cylinder) for validation. The students would build upon the previous exercises by applying a mesh resolution study and a variation of input parameters.

In the current set of exercises, the students were expected to perform all of the concepts discussed in the last bullet point during the fifth exercise. This approach provided to large a leap in understanding for many of the students, so the revised set of exercises will make the process smoother for most students.

Table 1. Student Feedback from the Study.

Question	Minimum Rating	Average Rating	Maximum Rating
1. How much time (in hours), did you spend on average for an EasyCFD assignment?	1	2.9	8
2. Rate your opinion of EasyCFD on the scale below. *[1:hated it, 10: loved it]*	1	3.8	7
3. Rate your opinion of the EasyCFD assignments on the scale below. *[1: worthless, 10: beneficial]*	2	5.7	8
4. Rate your opinion of the difficulty of EasyCFD assignments on the scale below. *[1: too easy, 10: too hard]*	3	6.9	9
5. Would your opinion of EasyCFD be different if a network licensing approach was used instead of the activation code approach we used?	Yes: 5 responses, No: 12 responses		
6. Did the EasyCFD exercises make it clear to you that you should not trust the results of engineering software without any validation? *[1: definitely not, 10: definitely]*	1	7.3	10
7. Did the EasyCFD exercises make you more likely to use software validation in the future? *[1: definitely not, 10: definitely]*	2	6.0	10
8. Did the EasyCFD exercises give you ideas as to how to approach learning new software? *[1: definitely not, 10: definitely]*	1	6.9	10
9. What influence did the EasyCFD exercises have on your approach to learning new software on your own? *[1: no influence, 10: lots of influence]*	1	5.1	9
10. Does having my answers to various HW problems give you confidence in your solutions. *[1: no gain in confidence, 10: major gain in confidence]*	3	8.4	10
11. Did following a tutorial give you confidence in your ability to use the EasyCFD code? *[1: no gain in confidence, 10: major gain in confidence]*	2	5.5	10

IV. Concluding Remarks

The study showed that teaching engineering students "how to teach themselves" will be met with resistance since it differs from conventional pedagogy. The exercises in the study did a good job of demonstrating the importance of validation, however they can be improved to help the students learn self-teaching of engineering software. For this reason, the exercises will be updated for continued use in future courses. The solutions to the five exercises may also act as tutorials for the other faculty in the department.

V. Bibliography

1. Garrett, S. L. (1996) The Polk County Courthouse: a $47 Million Iaq Disaster. Presented at *1996 ASEE National Conference*.
2. Schlager, N. *Breakdown: Deadly Technological Disasters* (Visible Ink Press, 1995).
3. Petroski, H. *Design Paradigms: Case Histories of Error and Judgment in Engineering* (Cambridge Univ. Press, 1994).
4. Petroski, H. *To Engineer is Human: The Role of Failure in Successful Design* (Vintage Press, 1992).
5. Ghosh, S. (2002) Humbleness As a Practical Vehicle for Engineering Ethics Education. *32nd Annual Frontiers in Education*, Vol. 3, DOI: 10.1109/FIE.2002.1158738.
6. Chiles, J. R. *Inviting Disaster: Lessons from the Edge of Technology* (HarperCollins, 2002).
7. Kececioglu, D. B.; Wang, W. (1997) On the Lower Confidence Limit of the Calculated Reliability for Mechanical Components and Structural Members. *Collection of Technical Papers – AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, Vol. 4, 2885-2889.
8. Thornton, J. (2011) No Testing Allowed: Nuclear Stockpile Stewardship is a Simulation Challenge. *Mechanical Engineering Magazine*, Vol. 133(5), 38-41.
9. Morsi, R.; Ibrahim, W.; Williams, F. (2007) Concept Maps: Development and Validation of Engineering Curricula. *ASEE Database*, paper AC2007-1314.
10. Lopes, A. M. G. (2010) A Versatile Software Tool for the Numerical Simulation of Fluid Flow and Heat Transfer in Simple Geometries. *Computer Applications in Engineering Education*, Vol. 18(1), 14-27.
11. http://ansys.com/Products/Simulation+Technology/Fluid+Dynamics/ANSYS+Fluent
12. http://www.cd-adapco.com/products/star_cd/index.html
13. http://openfoam.com/
14. http://www.cfd-online.com/Wiki/Validation_and_test_cases#2-D_test_cases