AC 2011-1957: USE OF FLUENT SOFTWARE IN A FIRST-YEAR ENGINEERING MICROFLUIDIC DESIGN COURSE

Barbara Elizabeth Carruthers, The Ohio State University

Barbara E. Carruthers is a Mechanical Engineering graduate student at The Ohio State University and a Graduate Teaching Assistant for the OSU Fundamentals of Engineering for Honors (FEH) Program. Ms. Carruthers with graduate with her M.S.M.E. from Ohio State in 2012.

Paul Alan Clingan, The Ohio State University - EEIC

Lecturer - First Year Engineering Program Engineering Education and Innovation Center The Ohio State University MS - Chemical Engineering - Bucknell University - 1988 BS - Chemical Engineering - Bucknell University - 1986

Use of FLUENT Software in a First-Year Engineering Microfluidic Design Course

Carruthers, B.E. and Clingan, P.A.

Abstract

Currently, the integration of computational fluid dynamics (CFD) software is typically only seen in higher level courses at the undergraduate level [1] [2] [3]. In this case, students are equipped with the basics of fluid dynamics from their core classes, which allows them to focus on the implementation of such problems and the rote mechanics of operating in the CFD environment [4]. This approach, while helpful in preparing students for industry, robs them of a visualization tool which could have supplemented traditional course material throughout their undergraduate careers. The Ohio State University has created a “cornerstone” design course, available to freshmen, in which basic micro-fluid dynamics concepts are presented, using CFD software as a visualization and verification tool [5]. This allows freshmen to identify and develop an interest in fluid dynamics at the start of their undergraduate career, perhaps shaping their progression throughout the curriculum. Overall, this course is still somewhat a work-in-progress, but also a unique proof of concept for teaching computational modeling early in an undergraduate curriculum. Results were seen in the high quality of the work produced by students, as well as the enthusiastic student reviews for the course. The course concludes with a poster presentation, judged by industry professionals in the fields of chemical, mechanical, civil, and aerospace engineering, as well as biology, chemistry, and medicine.

Introduction

Today, the use of simulation technologies is prevalent across the sciences. From the interactions of subatomic particles to orbital mechanics, computer simulation extends the reach of pure analytical models, decreases computation time, and provides verification to experimental results. And recently, increases in the speed, reliability, and availability of computing resources have amplified industry’s use of such methods.

As such, there is a high demand for college graduates who, in addition to their traditional analytical course work, have a strong foundation in computational methods, such as FLUENT or FLOW-3D [6] [1] for fluid modeling. However, the learning curve for computational modeling tools is high and understanding the math behind the programming requires a thorough understanding of higher-level mathematics. Additionally, across disciplines, the field of computer simulation is highly fragmented [7], making a cross-discipline teaching approach a complicated endeavor.

Educators have responded by including the computational modeling software in their undergraduate curriculum [8] [9], often focusing on degree-specific problems using a case-study approach. Typically, this is done in upper-level undergraduate classes, once students have mastered the analytical solutions to these problems and have been exposed to enough mathematics to understand the calculations performed by the computer [1] [2] [3]. This approach aims to ensures that students understand the full power of the tool, both in its capabilities and its limitations, from their first interaction [10] [11]. In this manner, students are familiar with the mechanics of the problem and instead, focus on the rote mechanics of operating the software [8].

However, problems can arise when teaching computational modeling tools in this manner. For one, a student's first interaction with computational modeling software may be outside of the classroom, such as in an internship setting, which may set the student up for frustration and failure. Previous research has shown that in the fields of aeronautical, mechanical, chemical, and biomedical engineering, fluid simulation tools are frequently too difficult for students to learn from tutorials or industry supervisors [12] [11]. And although internships typically increase retention [13] [14] [15], negative experiences can drastically affect student retention, especially the retention of women [16].

In addition, industry professionals are finding that a single year of computational modeling experience is simply too shallow to be of substantial use to a graduate intending to work with such tools professionally [17]. Also, the computational modeling experience students gain in capstone projects is frequently too difficult [18] or not representative of real-world problems [19]. This lack of experience puts a burden on employers [20], but also puts well-trained students in high demand.

Recent advances in personal computers have made computational modeling a possibility for students as a supplement to their studies. Waiting until upper-level courses to introduce computational modeling robs them of a valuable tool which could have enhanced their traditional course material throughout their undergraduate careers. In particular, student-level software is suitable for use as an industry proof-of-concept [21] [22]. Providing students with early access to computational modeling software has the potential to allow them to mirror this process, giving them a feel for the viability of their ideas: valuable feedback as they're learning the art of design.

Problem Formulation

The Ohio State University has developed an alternate approach to teaching computational modeling tools, specifically focusing on the introduction of computational fluid dynamics (CFD) software to a class of first-year honors engineering students.

A successful introduction meant overcoming some serious concerns regarding the educational background of the students [23], as well as determining an effective method for conveying complicated fluid mechanics concepts. It was the intention in the creation of the course to employ traditional teaching methods in the instruction of CFD, and many of these were retained from year to year. Since then, however, more out-of-the-box methods have arisen, sometimes as planned curriculum development and sometimes as ad-hoc solutions to unexpected problems. Here, we present the course as it stands today: still somewhat a work in progress, but also a unique proof of concept for teaching computational modeling early in an undergraduate curriculum.

Overall, through consistent course evaluations, quality of results, and industry responses, it was shown that teaching CFD software to first-year students can be beneficial in explaining fluid dynamics, especially at an early stage in a student's undergraduate career. Additionally, these results may have a broader context – implying that early introduction to complicated technology works in the classroom across the board.

Course Overview

As published previously, The Ohio State University offers a three course sequence to first-year honors engineering students, consisting of a drafting and computer aided drawing class, a programming class, and a "cornerstone" design class [5]. Students can choose from three options for the design class: brother-and-sister classes focusing on industrial design, robotics design, or nano- and microfluidics design. All three courses employ similar structures, based on a tried-and-true project-based learning approach [24].

The nano- and microfluidic design course consists of two concurrent parts. In the microfluidics portion, teams of four students learn the basics of flow in channels by reading and completing a guided example. Then, students are asked to create a MATLAB program in which a user could input any of several parameters and calculate the velocity distribution in a channel. In this assignment, students are meant to build on their MATLAB experience from the Winter Quarter programming class and focus on the calculation of fluid mechanics.

In a process designed to mirror graduate university research, this knowledge is put to use. Teams are asked to design unique microchannels to test the adherence of yeast to varying patterned surfaces. These channels are constructed and tested by the teams, who present their findings to a panel of industry and university experts [5].

Near the middle of the course (week 4), students gain some preliminary experience using CFD software through examples completed by students during class. Then they are asked to recreate their MATLAB results in CFD software, and predict a velocity distribution across their team's unique channel. If time allows, some students choose to model the flow over a square step, then a single yeast cell, and use this knowledge to predict the adherence of a yeast cell to a smooth channel.

Throughout the process, undergraduate teaching assistants who have completed the course in prior years help students navigate CFD software's steep learning curve [25]. This interaction supplements the instructor's lecture and provides students with frequent one-on-one support. These established teaching methods [26] [27] provided a consistent basis for the use of proven, but non-traditional teaching methods, such as the use of published research journals to supplement class materials and the use of electronic journals to anonymously receive student feedback throughout the course [28].

Overall, the microfluidics portion of the class consists of roughly half the class grade, including the chip construction, testing, and documentation. Out of a possible 1000 points, the MATLAB fluid mechanics program is worth 50 points (5%), and the CFD simulations are worth a total of 75 points (7.5%). Additionally, students are evaluated on their documentation, presentation skills, and teamwork.

Initial Conditions

The microfluidics portion of the class was designed to mitigate some of the concerns typically associated with the early introduction of CFD software [23] [29], specifically, unfamiliarity with basic fluid mechanics concepts, insufficient math background, and misuse of CFD software. And these concerns certainly applied to the first-year students enrolled in the course - the only assumed knowledge basis was a kinematics class through the Physics Department, some calculus, and the two preceding classes in the Fundamentals of Engineering for Honors series.

However, of these students, all were honors-designated students, and nearly all had above a 3.0 GPA. Since this was the third class of a three-course sequence, it could be assumed that all students had self-selected engineering, and had specifically identified an early interest in biomedical, chemical, or mechanical engineering.

Initial Fluid Mechanics Programming

To bridge the gap in basic fluid mechanics concepts, and guided by the requirements of upper-level CFD students in the Mechanical Engineering Department at The Ohio State University [30], the following points were identified as areas of probable concern in teaching CFD to first-year students:

1. Varying pressures and their role in fluid motion
2. Meaning and usage of the Reynolds Number
3. Viscosity and its role in pressure flow, lift, and drag
4. Role of the no-slip condition
5. Role of compressibility and when it’s negligable
6. Differences between 2-dimensional and 3-dimensional fluid flow
7. Use and meaning of streamlines
8. Methods of presenting CFD results

These concepts could be grouped into three categories: inviscid, incompressible pressure flow (points 1-2), extensions of inviscid, incompressible theory (points 3-6), and the graphical presentation of data (points 7-8).

Students were first exposed to a guided example, illustrating the effects of pressure flow on a fluid element (Figure 1), and then accounting for viscosity on the channel walls and compressibility effects.

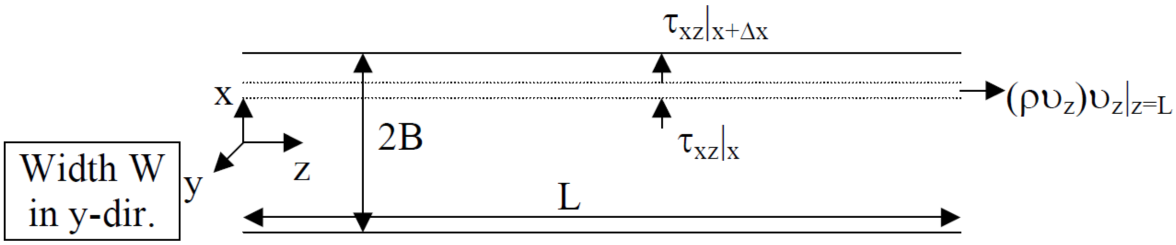

Figure 1: Fluid Element Example

Each student is required to translate the solution to the example into a MATLAB script. This program accepts all but one the following parameters, and solves for the missing parameter, as well as the output parameters.

Table 1: Required Input / Output Parameters

Input Parameters	Output parameters
volumetric flowrate	average velocity
channel dimensions	shear stress at the wall
change in pressure across the channel	Reynold's Number
viscosity	entrance length
channel cross-sectional shape	

These programs were evaluated by the teaching staff to determine how thoroughly students understood the material, and undergraduate teaching assistants provided review sessions and additional support to students who were performing poorly. Though a difficult assignment, each year, the majority of the students see the importance in getting acquainted with the material. In 2010, the average for the MATLAB programming assignment was an 80.6%.

In order to address point #8, the layout and creation of a research poster – as well as the proper presentation of CFD results – was presented in a guest lecture by a judge for the university's undergraduate research forum.

Initial Introduction of CFD to First-Year Students

Because of the incomplete math background of the students, it was impossible to teach the complicated higher-order calculations behind the CFD processes. However, an understanding of these calculations was crucial to the effective use of CFD software, even in the preliminary stages.

Instead of a complicated mathematical derivation, the first introduction to CFD software was a discussion. Since all of the students in the course were first-year students, and many were completing their calculus sequence during the course, Riemann sums were re-introduced and used as a basis for understanding – if not actually computing – numerical methods [31]. From this basis, students were introduced to the Navier-Stokes Equations (using words, since none of the students had taken a differential equations course) and together, the class talked-through how a computer might solve a problem like the one they had solved in MATLAB (Figure 2). Although this was far from allowing the students to begin to program their own CFD software, as in some upper-level CFD classes, students still gained appreciation for the complexity of the code and an understanding of the information needed by CFD software.

Extensive use of examples accompanied the introduction to CFD software. Students were asked to think about the parameters at work in a variety of familiar situations, such as a weight on a spring and a bending beam, and then were introduced to approximation techniques for solving for unknowns.

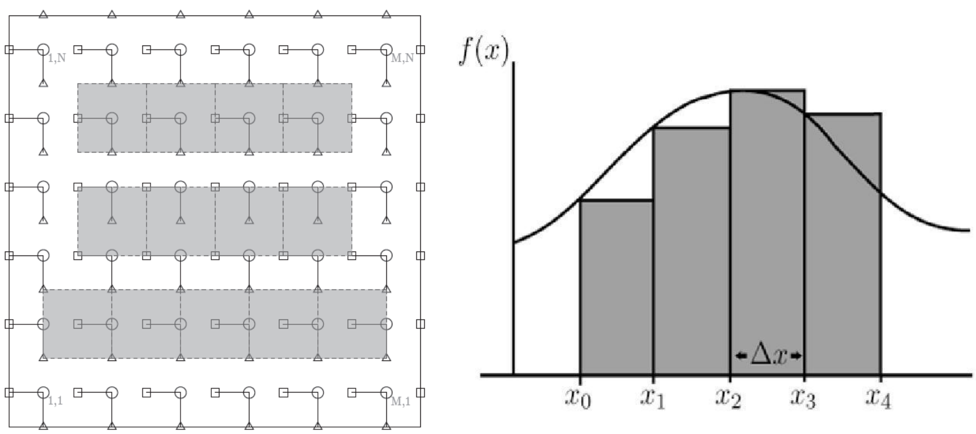

Figure 2: Comparison Images Used in the Discussion of Discretization [32]

Using this basis, students were then asked to complete two simple step-by-step tutorials created by the teaching staff, guiding them through the intricacies of working with meshing and CFD software. The first tutorial calculated the 2-dimensional flow through a channel and the second tutorial modified the channel to account for 3-dimensional effects. Students, once completing the tutorials were able to develop graphical representations of the shear stress pressure drop along the channel (Figures 3 and 4).

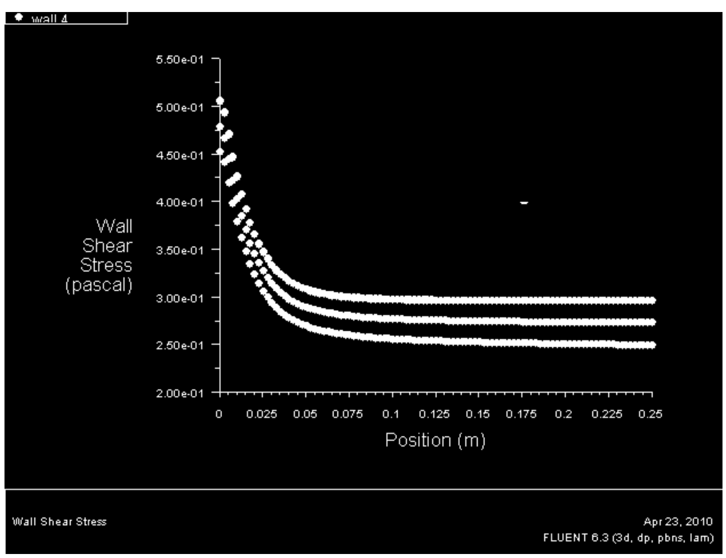

Figure 3: Shear Stress Results, 3D Case

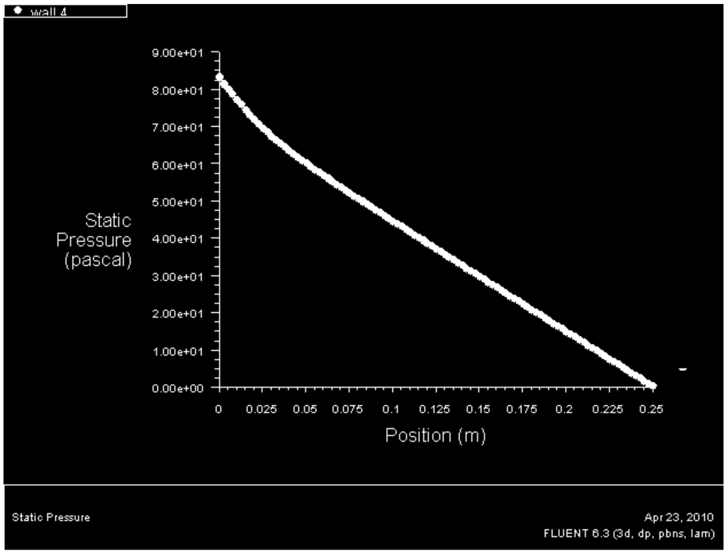

Figure 4: Pressure Results, 3D Case

Concerns with Open-Ended Design Using CFD Software at the First-Year Level

Once the students were familiar with the meshing and CFD environments, it became desirable to allow them to experiment with the tools and visualize the flow in channels of their own design. However, this brought to light the last common concern with teaching CFD in early undergraduate courses [23], still unaddressed in the course: the probable misuse of CFD software by inexperienced students.

In many ways, it was this last concern that was the most troubling. Part of the "art" of CFD software use is the knowledge of when a problem can be treated as a "black-box" [33] [34], so this wasn't a skill that could be taught to first-year students in the span of a ten-week course. The students were just learning the basics of fluid mechanics, so they had no intuition about what their results should look like, and due to their math background, it was impossible to completely illustrate when and how each parameter was used. It was an issue of minor consequence within the constraints of the class, but one with far reaching implications.

Is teaching CFD worth the class time, if students can't differentiate between an incorrect solution and a correct one in practice? For that matter, how can an educator determine when the students have the background to create accurate models of fluid flow? And what metric can be used to evaluate these crucial intangibles –intuition, experience, "art" – and when is it enough [35]?

In the end, the solution was simple. Students were provided with a list of problem-solving tactics, outlining when to use certain constraints, and when not to necessarily trust their CFD solutions [35] [36]. The instructional team attempted to bolster the intuition of the students by exposing them to an array of CFD projects from around campus, explaining the problem formulation behind them, and showing the first-year students what "good" results look like. Through coursework, modeling flow in channels became somewhat routine, and students learned what to expect from these types of problems.

In addition, a lecture was added to the course curriculum outlining the basic statistics required to state a confidence level for experimental results. Each student was required to write a MATLAB script calculating a significance level for each confidence level. These programs were used to state the confidence level of their experimental results. The course structure itself was helpful in the discussion of validation and verification of CFD results – by solving for experimental and computational results in parallel, students were able to identify potential pitfalls of using either approach as the "correct" solution.

Open-Ended Design Using CFD Software

The CFD portion of the class concluded with the use of CFD software to visualize the 3-dimensional flow in channels of their own design. Here, students worked in their teams of four to create a unique design for a microfluidic channel suitable for testing the adherence of yeast particles to a patterned surface. These channels required an input and output for both yeast and water, as well as a pre-defined "testing region" in which the flow could be considered laminar (Figure 5).

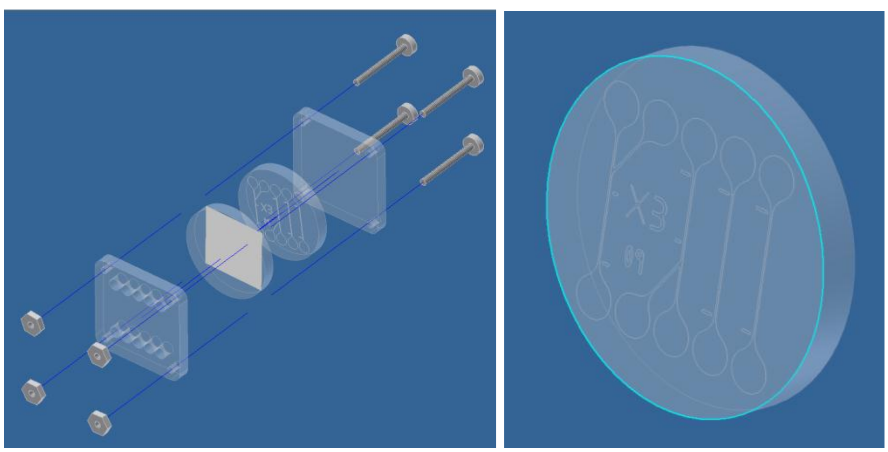

Figure 5: Student-Designed Microfluidic Chip and Holder (2009)

Ideally, the instructional team had hoped to see teams using CFD software as a proof-of-concept of potential designs. However, time constraints only allowed the students to create one CFD representation of the flow in their unique channel, usually after it was constructed.

Still, students were able to use CFD software to accurately model the flow in a unique channel; the average grade for the 3-dimensional mesh and converged CFD solution was 130% (with extra-credit) and 86.76%, respectively (using 2010 data). These results were included in the teams' final reports, as well as on research posters describing their conclusions (Figures 6 and 7).

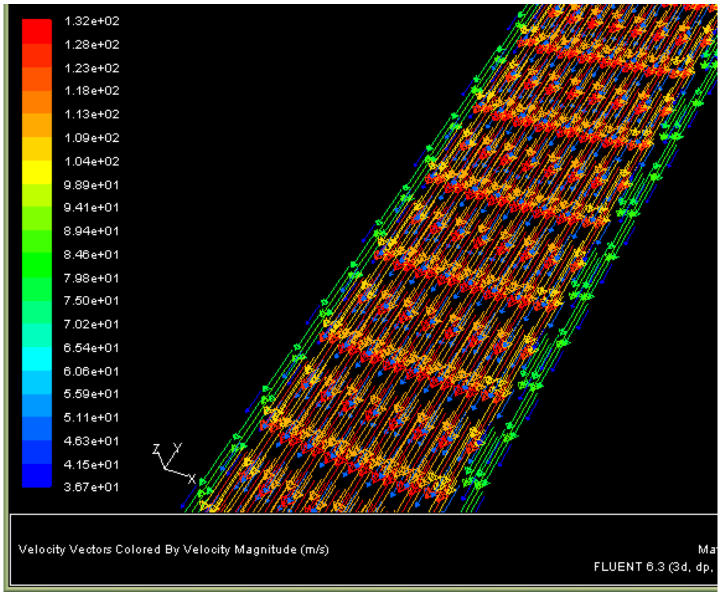

Figure 6: Velocity Distribution as Presented by Students (2009)

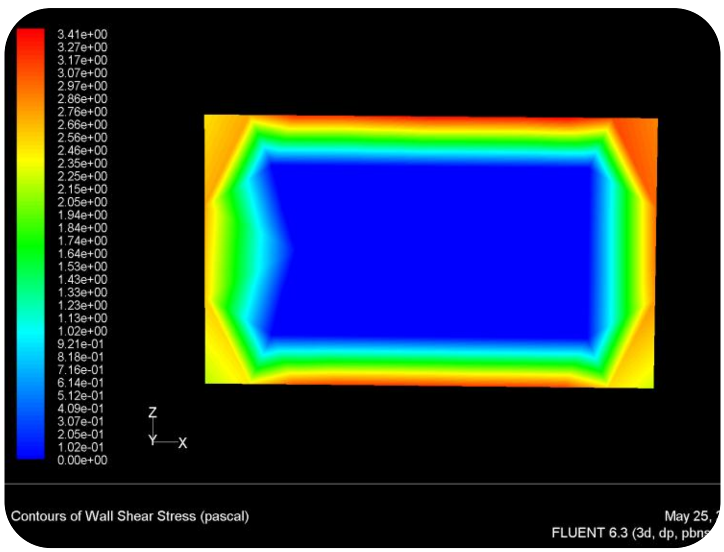

Figure 7: Shear Stress Distribution as Presented by Students (2009)

Discussion / Evaluation of Progress

As seen in recent literature, classes such as the one developed at The Ohio State University can excite undergraduate students and motivate them to do well in their core engineering classes [(4)]. Although this class consisted of first-year students, it was no exception. This was evident in the course evaluations and weekly journal entries collected from students.

Some excerpts from these responses include:

- "… it's nice have experience with MATLAB, GAMBIT, and FLUENT." (2010)

- “I feel that I have had many advantages over this past year [due to this student’s involvement in the program]. I believe that the inventor, autocad, programming and the design projects give me a huge upper hand compared to other students. I have also learned the basics in Fluent and Gambit. This would only help me down the road in my future.” (2009)
- “disadvantage: receiving a worse GPA; advantage: great opportunities using real-life engineering programs i.e. FLUENT, GAMBIT, C, C++, etc..” (2009)
- “One main advantage I feel that FEH students have is the exposure we've had to so many different things that first year students typically would not have the chance to see (i.e. FLUENT from [the course described in this paper]).” (2009)

At the aggregate level, students responded well to the instruction in CFD software upon reflection, but sometimes felt too busy during the course. Additionally, a year after the completion of the course, some students noted that their use of CFD software helped them gain confidence in their core classes, as well as identify personal strengths and weaknesses within their field.

The first-year students in this class also had the advantage of having their CFD results critiqued by industry professionals and professors from across campus. The evaluations from these judges are averaged to determine a team’s score. In 2010, the average grade was an 86.34% - indicating that the majority of the judges found the CFD work presented by the students on par with undergraduate research in their field.

At this point in time, CFD is too recent an addition to the course to be able to discuss viable retention data or to determine whether the early introduction of CFD software is beneficial throughout the undergraduate careers of students. Future work includes the quantification of the effects of the early introduction of CFD software on the undergraduate coursework of graduating seniors.

Concluding Remarks

Taken as a proof of concept, this microfluidic design class illustrates the possibility of introducing CFD software as early as the first year. However, this course was never intended to replace the higher-level curriculum of the students, but rather to be a first introduction to the material. Although students may not initially have the fluid mechanics or math backgrounds to work with CFD software, in ten weeks, this course was able to bridge the gap and provide them with enough of a basis to use this software as a visualization tool throughout their undergraduate careers. Even in industry, CFD results need validation and verification [35], and if educators begin to reinforce these ideals with the introduction of CFD software, students will begin to treat them as an integral part to CFD work, and research as a whole.

Works Cited

1. *A Course in Micro- and Nanoscale Mechanics.* **Crone, W.C.** s.l. : American Society of Engineering Education, 2010.

2. *Integration of Simulation into the Undergraduate Fluid Mechanics Curriculum Using FLUENT.* **Bhaskaran, R. and Collins, L.** s.l. : American Society of Engineering Education , 2003.

3. *A Three-Semester Interdisciplinary Educational Program in Microsystems Engineering.* **Ameel, T., Gale, B., and Harvey, I.** s.l. : American Society of Engineering Education, 2003.

4. *Using Computational Fluid Dynamics to Excite Undergrad Students about Fluid Mechanics.* **Pines, D.** s.l. : American Society of Engineering Education, 2004.

5. *Work in Progress: Micro-/Nanotechnology 'Lab on a Chip' Research Project for First-Year Honors Engineering Program.* **Clingan, P.A., Tomasko, D.L. and Allam, Y.** s.l. : Institute of Electrical and Electronics Engineers, 2006. Frontiers in Education Conference.

6. *Education and Training of the Nanotechnology Workforce.* **Fonash, S.** s.l. : Journal of Nanoparticle Research, 2001.

7. **Zeigler, B.P.** *Theory of Modeling and Simulation.* Malabar : Krieger Publishing Company, 2000.

8. *Using Computational Fluid Dynamics to Excite Undergraduate Students About Fluid Mechanics.* **Pines, D.** s.l. : American Society of Engineering Education, 2004.

9. *The Use of Advanced Simulation Tools in Capstone Design Projects.* **Barber, T.** s.l. : World Scientific and Engineering Academy and Society, 2007.

10. *Use of Computational Fluid Dynamics (CFD) in Teaching Fluid Mechanics.* **Sert, C. and Nakiboglu, G.** s.l. : American Society of Engineering Education, 2007.

11. *Computational Aerodynamics Goes to School: A Course in CFD for Undergraduate Students.* **Cummings, R. and Morton, S.** s.l. : American Institute of Aeronautics and Astronautics, 2005.

12. *Hands-On CFD Educational Interface for Engineering Courses and Laboratories.* **Stern, F. et. al.** s.l. : American Institute of Aeronautics and Astronautics, 2006.

13. *Improving Engineering Undergraduate Retention via Research and Internships.* **Seevers, M. et. al.** s.l. : American Society Engineering Education, 2006.

14. *An Innovative Method to Reaslistically Track Engineering Student Retention and Academic Progress.* **Pyke, P. et. al.** s.l. : American Society of Engineering Education, 2007.

15. *Diversity Programs and Nuclear Engineering Education: Outreach, Retention, and Engagement.* **Marshall, L. and Bourham, M.** s.l. : American Society of Engineering Education, 2007.

16. *Work in Progress - A Study of How Real-World Engineering Experience Can Affect Women's Academic Career.* **Walter, D. et. al.** s.l. : Institute of Electrical and Electronics Engineers, 2008. Frontiers in Education Conference .

17. *The Use of CFD in Capstone Senior Design Projects.* **Barber, T.J.** s.l. : American Institute of Aeronautics and Astronautics, 2008.

18. *Equipping Engineers to Solve Computational Fluid Dyanmics Problems in Senior Design: Challenges and Successes Using a Commercial CFD Packade.* **Burban, P.M. and Harwood, A.H.** s.l. : American Society of Engineering Education , 2006. Annual Conference.

19. *Chalmers Program in Turbulance: A Proposed Model for Higher Level Undergraduate Education in General.* **Johansson, G.T. and George, W.K.** s.l. : Institute of Electrical and Electronics Engineers, 2005. Information Technology Based Higher Education and Training.

20. *CFD Advanced Professional Education via the Internet.* **Baker, A.J. et. al.** 2005 : American Society of Mechanical Engineers. International Mechanical Congress and Exposition.

21. *Disposable SMART Lab on a Chip for Point-of-Care Clinical Diagnostics.* **Ahn, C.H. et. al.** s.l. : Institute of Electrical and Electronics Engineers, 2004.

22. *Imeplementation of a Continuous Adjoint for Topology Optimization of Ducted Flows.* **Othmer, C.** s.l. : American Institute of Aeronautics and Astronautics, 2007. Computational Fluid Dynamics Conference.

23. *The Role of Case Studies in CFD Education.* **Stubey, G.D. and Hutchinson, B.R.** s.l. : American Society of Mechanical Engineering , 2002.

24. *Enhancing the Engineering Curriculum Through Project-Based Learning".* **Hadim, H.A. and Esche, S.K.** s.l. : Institute of Electrical and Electronics Engineers, 2002. Frontiers in Education Conference.

25. *Hands-On CFD Education Interfaces for Engineering Courses and Laboratores.* **Stern, F. and Xing, T.** s.l. : American Society of Engineering Education, 2006, Journal of Engineering Education .

26. *Teaching Non-Ideal Reactors with CFD Tools.* **Madiera, L. et. al.** s.l. : American Society of Engineering Education, 2004.

27. *Comparison of Student Learning in Physical and Simulated Unit Operations Experiements.* **Wiesner, T.F. and Lan, W.** 3, 2004, Journal of Engineering Education, Vol. 93, pp. 195-204.

28. *The Effectiveness of Learning Simulations for Electronic Laboratories.* **Cambell, J.O. et. al.** 1, s.l. : American Society for Engineering Education, 2002, Journal of Engineering Education, Vol. 91, pp. 81-87.

29. *Introducing Natural-Convective Chilling to Food Engineering Undergraduate Freshmen: Case Studied Assisted by CFD Simulation and Field Visualization.* **Rabi, J.A.** 1, s.l. : Wiley Publications, Comupter Applications in Engineering Education, Vol. 17, pp. 34-43.

30. AEROENG 615 - Introduction to Computational Aerodynamics. *The Ohio State University College of Engineering.* [Online] 2010. [Cited: January 18, 2010.] http://mae.osu.edu/course/aeroeng/615.

31. **Toro, E.F.** *Riemann Solvers and Numerical Methods for Fluid Dynamics.* Berlin : Springer Press, 2009.

32. s.l. : The Ohio State University Mechanical Engineering Department.

33. *Mode Pursing Sampling Methods for Discrete Variable Optimization on Expensive Black-Box Functions.* **Sharif, B. et. al.** 2, s.l. : American Society of Mechanical Engineering, 2008, Journal of Mechanical Design, Vol. 130. 021402 .

34. **Carle, A. et.al.** *Applications of Automatic Differentiation in CFD.* s.l. : American Society of Aeronautics and Astronautics , 1994. ANL/MCS/CP--82980; CONF-940662--1.

35. *Toward Establishing Credibility in Computational Fluid Dynamics Simulations.* **Rizzi, A. and Vos, J.** 5, s.l. : American Institute of Aeronautics and Astronautics, 1998, AIAA Journal, Vol. 36, pp. 668-675.

36. **Bennett, J.** *GAMBIT/FLUENT Checklist.* First-Year Engineering Program, The Ohio State University. Columbus, Ohio : s.n., 2007. White Paper.

37. *Assessment by Peers - An Effective Learning Technique.* **Dziedzic, M. et. al.** s.l. : Institute of Electrical and Electronics Engineers, 2008. Fronteirs in Education Conference.

38. *Accuracy and Reproducability of CFD Predicted Wall Shear Stress Using 3d Ultrasound Images.* **Augst, A.D. et. al.** 2, s.l. : American Society of Mechanical Engineering, 2002, Journal of Biomechanical Engineering, Vol. 125.