2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering Education for 130 Years
Paper ID #39352

Impact of Computation in Undergraduate Curriculum : Alumni Perspective

B. Ruşen Argun, University of Illinois at Urbana - Champaign

Ruşen is a Ph.D. student in the department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign. He is the computational teaching assistant of the Materials Science and Engineering department for academic year 2022-2023, and participated in the teaching activities about computations in several undergraduate courses. His research is about coarse-grained simulations of soft materials. He enjoys sailing when the weather is nice.

Prof. Andre Schleife,

AndrÃ© Schleife is a Blue Waters Assistant Professor in the Department of Materials Science and Engineering at the University of Illinois at Urbana-Champaign. He obtained his Diploma and Ph.D. at Friedrich-Schiller-University in Jena, Germany for his theo

Dr. Cecilia Leal

CecÃlia Leal is an Assistant Professor in the Department of Materials Science and Engineering and the Frederick Seitz Materials Research Laboratory at the University of Illinois, Urbana-Champaign since 2012. She graduated in Industrial Chemistry from Coi

Prof. Dallas R. Trinkle, University of Illinois at Urbana - Champaign

Dallas R. Trinkle is an associate professor in Materials Science and Engineering at Univ. Illinois, Urbana-Champaign. He received his Ph.D. in Physics from Ohio State University in 2003. Following his time as a National Research Council postdoctoral resea

Prof. Nicola H. Perry, University of Illinois at Urbana - Champaign

Nicola Perry has been an assistant professor in Materials Science and Engineering at UIUC since 2018. She received her PhD degree in the same field from Northwestern University (NU) in 2009 and subsequently held postdoctoral positions at NU, Kyushu University (Japan), and MIT. Prior to joining the faculty at UIUC she was a World Premier Initiative assistant professor in the International Institute for Carbon-Neutral Energy Research at Kyushu University and a research affiliate at MIT. Her research focuses on design and discovery of point-defect mediated active behavior in solid-state ionic materials for high- and low-temperature energy conversion and storage applications. Her teaching encompasses functional ceramics (senior-level class), solid-state ionics (graduate-level class), and phases and phase relations (sophomore-level class). She continues to support the incorporation of diverse computational modules in her courses. She is also active in educational outreach to middle-school and high-school students as well as the broader community on the topics of materials for sustainability, electrochemistry, and energy conversion, developing and leading interactive lessons and hands-on demos.

Prof. Jessica Krogstad, University of Illinois at Urbana - Champaign

Jessica A. Krogstad is an associate professor in the Department of Material Science and Engineering at the University of Illinois, Urbana-Champaign. She received her PhD in Materials at the University of California, Santa Barbara in 2012. Between 2012 and 2014, she held a postdoctoral appointment in the Department of Mechanical Engineering at Johns Hopkins University. Her current research explores the interplay between phase or morphological evolution and material functionality in structural materials under extreme conditions. She also maintains interest in engineering education, specifically in outreach and design thinking.

Dr. Matthew D. Goodman, University of Illinois at Urbana - Champaign

Dr. Goodman received degrees in Materials Science and Engineering from Iowa State (B.S. & M.S.) and the University of Illinois (Ph.D.). He is a lecturer in the Materials Science and Engineering Department at University of Illinois since 2014 with curr

Antonia Statt
Marie Agathe Charpagne

Impact of Computation in Undergraduate Curriculum : Alumni Perspective

Abstract

Computational methods have gained importance and popularity in both academic research and industrial applications in recent years. Since 2014, our team has consistently worked on reforming our Materials Science and Engineering curriculum at the University of Illinois Urbana-Champaign by incorporating computational modules into all mandatory undergraduate courses. Here, we investigate the impact and effectiveness of these computational modules in light of our recent graduates' feedback. We surveyed alumni who graduated between 2017 – 2021 and asked them about the benefits of the computational curriculum and the significance of computation for their career. "data analysis" was reported to be the most significant computational practice, followed by "programming" and "simulation tools". Python is the most prevalent programming language, and half of the respondents have reported to use it for their work. Particle based simulation tools are rarely used by our alumni, whereas continuum methods are more relevant, especially for alumni in industry. Graduates who pursued Ph.D. or Master's degrees benefited more from the existence of computational modules and would also benefit the most from qualitative improvements of the modules. Alumni have reported limited benefits of computational modules during their job search, but note a slightly positive impact on their job performance. Overall, our Alumni think that the current amount of computational material in the curriculum is ideal, but further analysis indicates there is still room for qualitative improvements. We find the perspective provided by alumni to be a valuable tool to evaluate the computational reform of the MatSE curriculum at the University of Illinois Urbana-Champaign and it is a useful guide on how to reshape and improve its effectiveness further.

Introduction

Computational methods in Materials Science and Engineering (MSE) are now essential in both research and industry. Results from surveys conducted in 2009 [1] and 2018 [2] showed that employers in the MSE field highly value computational materials science education and aim to hire 50% of their employees with some computational MSE background. As a response to the growing importance of computation in MSE, the curriculum of MSE at the University of Illinois Urbana-Champaign has been reformed by incorporating computational modules into the required core classes over the past years [3]. Detailed information about the computational modules added can be found in our previous publications [4, 5, 6], where our team performed several student surveys in order to spot the strengths and weaknesses of the newly introduced computational modules [7, 8, 9, 10]. Although student surveys are an effective tool when assessing the quality of the curriculum [11], student feedback lacks the perspective that the postgraduate experience provides [12]. Alumni surveys can be used as a guide to reshape a curriculum, especially for making it more relevant to the needs of employers [13]. Faculty members tend to hold the opinions of alumni in high regard, since graduates are more informed about industry or academia needs compared to current students [14].

In this study, we evaluate the impact of the computational reform of the curriculum by performing a survey of the recent alumni of our MSE department. We identify the weaknesses and potential improvements that could be made on computational modules in light of the graduates' feedback.

Survey Respondents

It is suggested that the alumni surveys performed with the aim of improving and evaluating the curriculum should be limited to recent graduates since their memories are still fresh, and university education plays a more significant role in the early years of an alumni's career [14]. On the other hand, alumni with more experience can provide a more mature perspective about the needs of the industry or academia compared to a fresh graduate. With these considerations in mind, the survey was sent out to the alumni that have graduated between 2017 and 2021. The majority of the respondents had at least some of the computational modules in their curriculum, since most of them were first introduced between 2014 and 2015. After that, some additional computational materials were added to the curriculum, however those were minor compared to the first set. As a result, we do not attempt to differentiate the exact amount of computational materials the respondents were exposed to, and only record whether they had any.

The respondents are divided into groups as seen in table 1 to facilitate the analysis of the survey data. Roughly three quarters of our respondents have some industry experience, whereas the rest is pursuing a Masters/Ph.D. degree without industry experience yet. Only 5 of the 38 respondents have reported to have no computational modules in their curriculum. Since this is a small sample size to obtain meaningful conclusions, we do not attempt to make any comparisons with the other groups, but we are still reporting their answers separately. We also note that one respondant without computational modules had a Ph.D. degree so their answers were grouped under "Masters/Ph.D." category instead of "No Computational Module" as can be seen in table 1.

Table 1: Distribution of the respondents to the survey. Answers were analyzed as 3 groups. All of the figures use the same colors for the 3 groups.

Sample Distribution	**Had Computational Modules**		Total
	Yes	No	
Industry	**25**	**4**	29
Masters/Ph.D.	**8**	**1**	9
Total	33	5	**38**

The range of computational competencies is wide. In this survey, we have divided computational skills into three groups: Use of simulation tools, use of programming languages and data analysis. Even though there is overlap between these groups, this classification still provides more precise conclusions for our purposes.

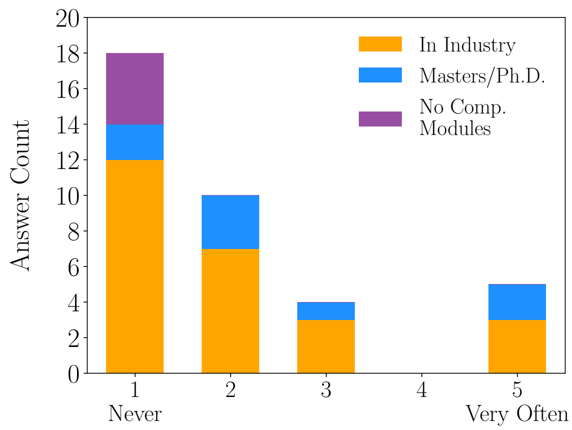
Simulation Tools

Our core curriculum currently introduces the computational methods: molecular dynamics (MD), density functional theory (DFT), "Computer Coupling of Phase Diagrams and Thermochemistry" (CALPHAD)[15], and finite-element analysis (FEA). The two following questions were asked specific to simulations tools:

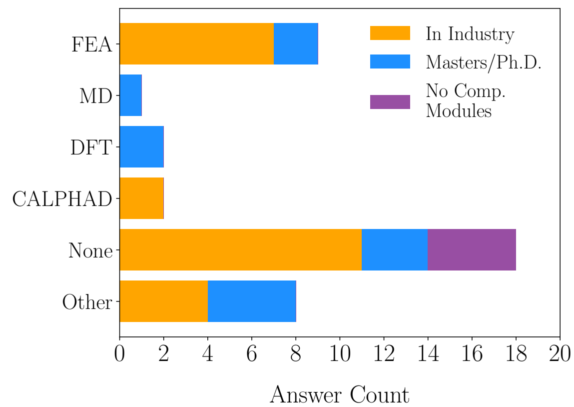
- Q1 : "How often do you use computational/simulation tools for your job?" (Never - ... - Very Often, 5 point Likert Scale)
- Q2 : "Which tools/methods are you using? Please check all that apply." (see Figure 1b for options)

Almost 50 % of the respondents reported that they never use simulation tools for their work, as seen in Figure 1a. Particle simulation tools like Density Functional Theory (DFT) and Molecular Dynamics (MD) were not used by any respondents in industry, however, a small number of participants pursuing masters or Ph.D. degrees have reported using them. The most common simulation tool is by far Finite Element Analysis (FEA) followed by CALPHAD that is rarely used by alumni in industry. In addition, 6 of the 25 answers to the open-ended question: *What are the topics that are essentially effective for your career in computational MatSE curriculum?* and 4 of 21 answers to the question: *What are the topics that were missing for your career in computational MatSE curriculum?* mention FEA, underlining its importance in industry for MSE. Similarly, Ref.[1] points out that at universities mostly particle based simulation tools are taught, whereas in industry continuum based models are more relevant. According to the survey from the same study, FEA is the most cited simulation tool used by MSE employers. The survey results suggest that students might benefit from more FEA related content in the modules, since it is commonly used and reported to be both an effective and missing part of the computational curriculum.

Even though DFT and MD is not used and CALPHAD is not common in industry, less than 10% of the alumni in industry think that there should be less computational material in the curriculum, as highlighted in Figure 5a. One possible interpretation of this apparent contradiction is that the utility in industry is not the only reason for the demand of computational modules. Tools like MD and DFT can provide a wider perspective on how to approach problems in MSE, they can help with understanding other theoretical concepts of the class content [4], or may simply be an interesting and engaging additional learning tool to the classical way the class content is covered. The question about the amount of computational content in the curriculum bundles the three groups of computational skills together. An alternative explanation is that students think coding and data analysis are the main cause of satisfaction with the current amount of computational classes, instead of the simulation tools. Overall, we think that a high level introduction of MD-DFT-CALPHAD is ideal as a part of the core curriculum and conclude any extra additional time should be allocated towards FEA methods, as seen in Figure 1b.



(a) How often do you use computational/simulation tools for your job?



(b) Which tools/methods are you using? Please check all that apply.

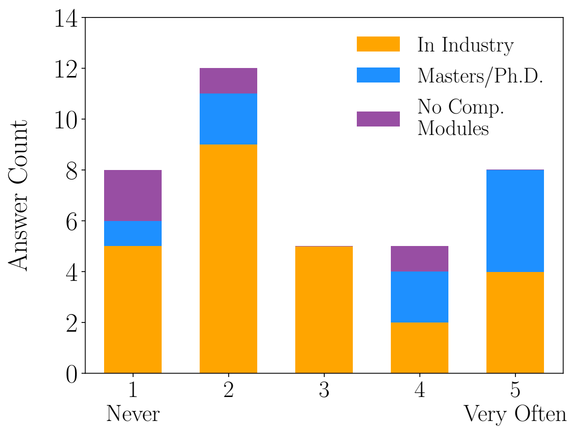
Figure 1: Simulation Tools (38 respondents were grouped into 3 categories. *In industry*: Respondents with industry experience that reported to have computational modules in their curriculum. *Masters/Ph.D.*: Respondents pursuing/have a Masters/Ph.D. degree without industry experience yet. *No Comp. Modules*: Respondents that have reported to have no computational modules in their curriculum.)

Programming Languages

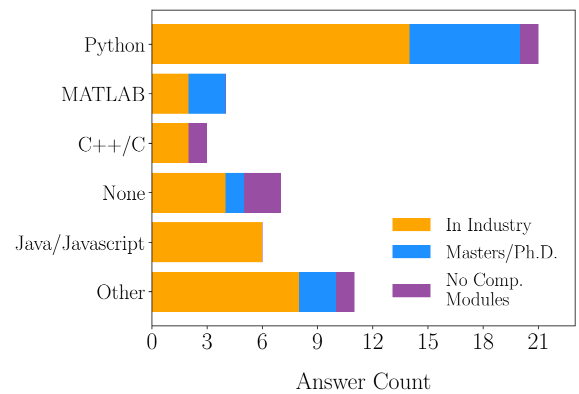
The following two questions were asked specific to the use of programming languages:

- Q3: "How often are you working with programming languages for your job?" (Never - ... - Very Often, 5 point Likert Scale)
- Q4: "Which programming languages are you working with? Please check all that apply." (see Figure 2b for options)

75 % of the alumni have reported that they program for their work more often than *never*, highlighting the significance of coding skills in the field. It appears that programming tasks are more common compared to performing simulations. As shown in Figure 2b, Python is the most prevalent programming language by far, since more than half of the respondents use Python for their work. In our curriculum, we have been phasing out MATLAB in favor of Python, and Figure 2b clearly supports that decision. Roughly 20% of the graduates are working with either one of C/C++/Java/Javascript, indicating that coding with compiled programming languages is also relevant for MSE graduates. Additionally, a study reported that switching from FORTRAN to MATLAB resulted in a drop in general coding skills of mechanical engineering students [16]. From our data and other studies in the literature [16], it is worth considering if a compiled language programming course should be part of the core curriculum of MSE. Among the programming languages in the "other" group, SQL stands out with 6 respondents (*i.e.,* 15%) using it, signaling the relevance of data bases and data analysis, which is discussed in the next section.



(a) How often are you working with programming languages for your job?



(b) Which programming languages are you working with? Please check all that apply.

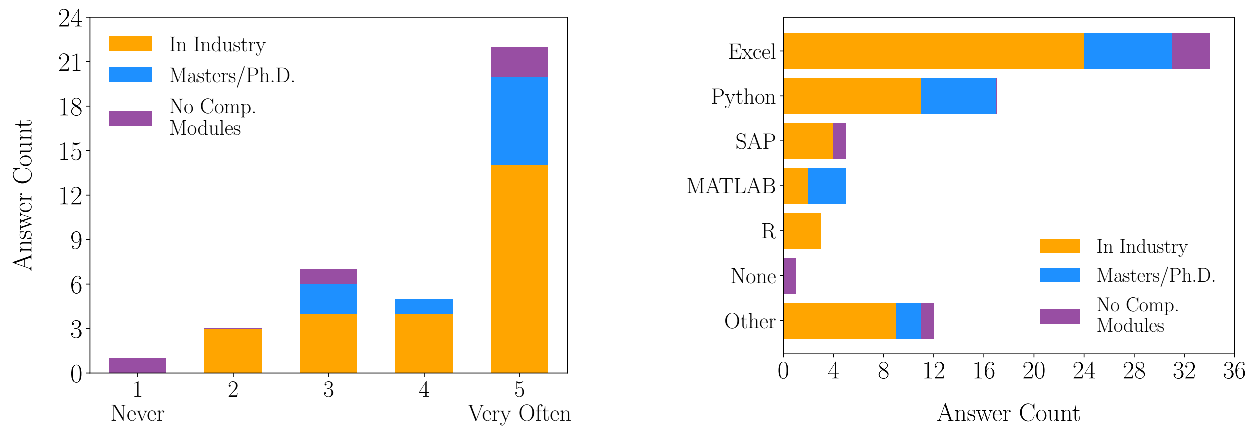
Figure 2: Programming Languages

Data Analysis

Among the three categories of computational skills we surveyed, data analysis is by far the most relevant one to our alumni. The following questions were asked specifically about data analysis:

- Q5: "How often do you perform data analysis for your job?" (Never - ... - Very Often, 5 point Likert Scale)
- Q6: "Which tools/languages are you using for data analysis? Please check all that apply." (see Figure 3b for options)

While nearly all of the respondents (37/38) are performing data analysis for their job, more than half of them are doing it *very often* as displayed in Figure 3a. 28% (7/25) of the respondents cite data analysis as one of the topics in computational MatSE curriculum that were essentially effective for their career. In addition, 28% (6/21) of the participants think that data analysis or skills that are closely related to it are missing in the computational MatSE curriculum. Graduates find data analysis material in the curriculum to be both effective and insufficient. Since most of the computational modules already have a data analysis part, this result strongly suggest that our current students may benefit from increasing the weight of data analysis in the curriculum. Given the prevalence of Python and data analysis, a list of relevant skills can be targeted systematically throughout MatSE computational curriculum to make students comfortable with Python and data analysis when they graduate. Among other data analysis tools JMP and Minitab are each cited 4 times.



(a) How often do you perform data analysis for your job?

(b) Which tools/languages are you using for data analysis? Please check all that apply.

Figure 3: Data Analysis

Skills related to Coding

In addition to "Programming Languages" section, to investigate the need for coding related skills in more detail, we asked alumni to rate how relevant the following practices for their job are (each on a 5-point Likert Scale):

- Skill 1: "Working with big coding projects collaboratively"
- Skill 2: "Managing workflows and version control"
- Skill 3: "Testing and verifying code"
- Skill 4: "Finding your way around complicated chunks of code"

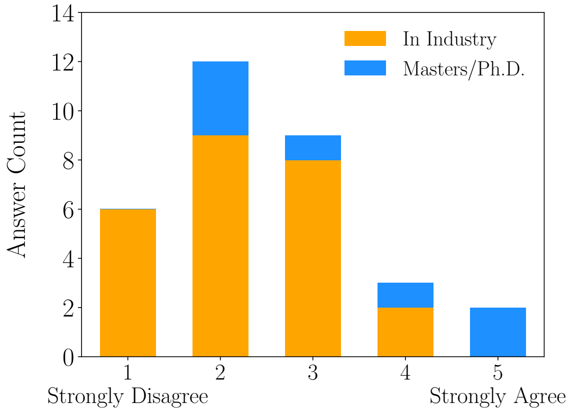
These practices can be considered to be specific to software engineers, however, in Figure S1 we show that for approximately half of the MSE graduates, they are more relevant than *not relevant at all*. The computational modules in our curriculum currently do not address any of these competencies, as they are not regarded as essential for a materials scientist. Additionally, they would require a solid background in programming before being introduced. Our results suggest that these skills are still not critically relevant to MSE graduates. We did not find any data or other studies from past years compare with, however, we speculate that relevance of these skills for a MSE graduate may increase in the future. Current results can be interpreted as an early warning that we should continue to monitor the trends in order to be ready to respond to the changing needs in industry and academia by reshaping the curriculum when needed.

Impact on Career

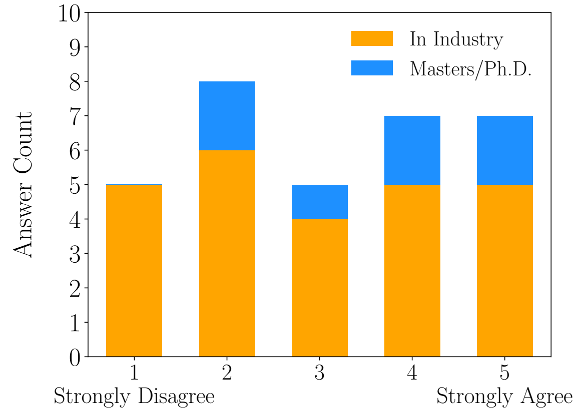
To assess the impact of computational curriculum on their career, participants were asked to rate the following statements on how much they agree with them:

- Statement 1: “The computational part of the MatSE curriculum helped me getting the job I wanted.”
- Statement 2: “The computational part of the MatSE curriculum helped me performing better at my job.”
- Statement 3: “The computational part of the MatSE curriculum has expanded my career opportunities.”
- Statement 4: “The computational part of the MatSE curriculum has an impact on my career choices.”
- Statement 5: “I could confidently apply for jobs requiring computational skills when I graduated.”
- Statement 6: “I think I have missed several opportunities due to lack of computational skills.”

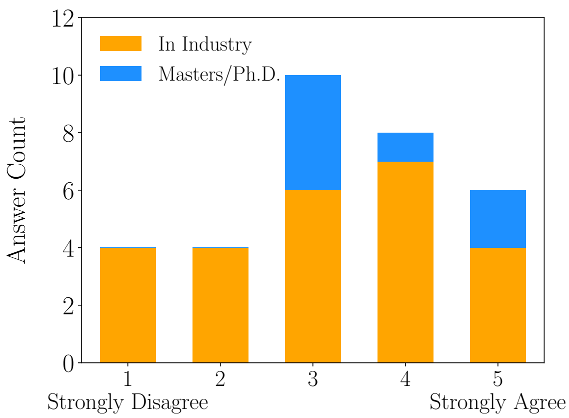
The results are plotted in Figure 4. On average, alumni think that the computational part of the curriculum was not helpful with getting the job they wanted, as shown in Figure 4a. Considering all the factors involved in landing a job, this result is not surprising. However, there is a slightly positive impact of the computational modules on their self-evaluated performance once the graduates get a job as displayed in Figure 4b. They slightly agree that computational materials have expanded their career opportunities. At the same time, most of them still do not feel confident applying to jobs that require computational skills. Their indicated lower confidence in applying for computational jobs is directly contradicting the strong agreement in that they did not miss any opportunities due to the lack of computational skills. This contradiction in Figure 4b can be interpreted in two ways: First, there were not many opportunities that could have been missed due to lack of computational skills; second, since they have acquired sufficient computational skills they did not miss opportunities. They also reported that there was a slight expansion of career opportunities thanks to computational skills (Figure 4c), thus the first interpretation cannot be the only one, and the second line of thinking must be valid to some degree. Overall, in Figures 4(c,f) we observe a net positive impact of computational modules on a graduates’ career. The rest of the data in Figure 4 indicate a more neutral impact on average.



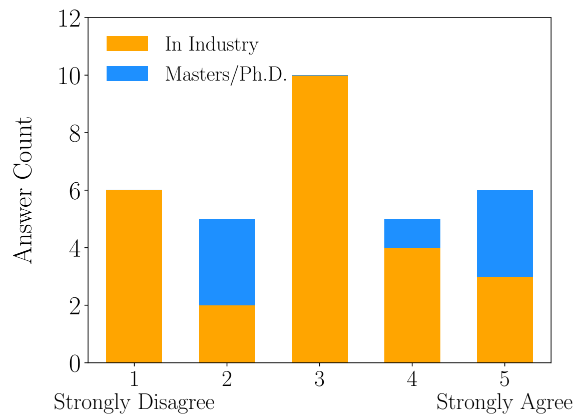
(a) The computational part of the MatSE curriculum helped me getting the job I wanted.



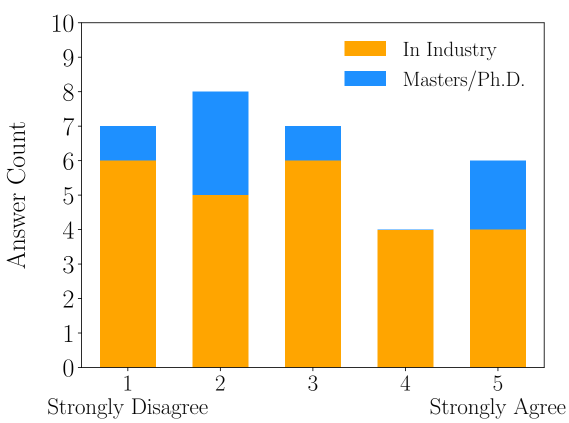
(b) The computational part of the MatSE curriculum helped me performing better at my job.



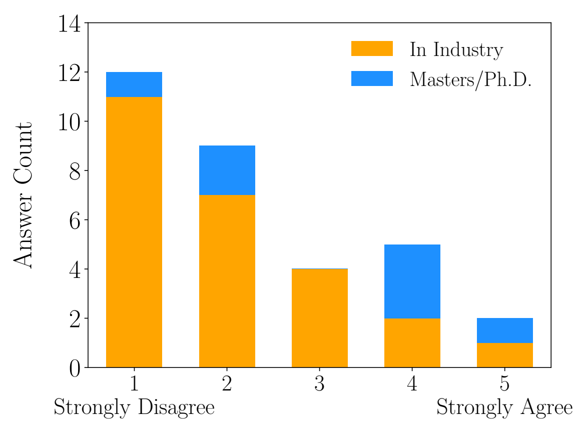
(c) The computational part of the MatSE curriculum has expanded my career opportunities.



(d) The computational part of the MatSE curriculum has an impact on my career choices.



(e) I could confidently apply for jobs requiring computational skills when I graduated.



(f) I think I've missed several opportunities due to lack of computational skills.

Figure 4: Impact On Career

Industry vs. Masters/Ph.D.

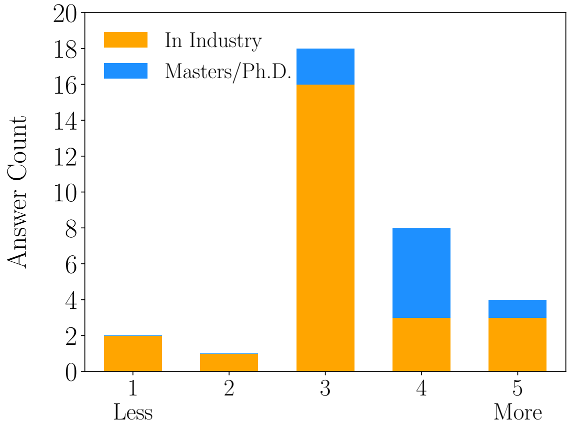
The number of respondents with no industry experience or equivalently pursuing Masters/Ph.D. degrees is 9. One should keep in mind that this is a small sample size throughout the following comparison between them and the respondents from industry.

There are significant differences in distribution of answers between industry and Ph.D./Masters careers. Alumni in the Ph.D./Masters route think that they benefited more from computational materials in expansion of career paths and in getting the job they wanted compared to the ones in industry, as shown in Figure 4(a,c). On average they report a higher impact of computation on their career (Figure 4d). Overall, this suggest that graduates in Ph.D./Masters programs benefit more from the computational curriculum than graduates in industry.

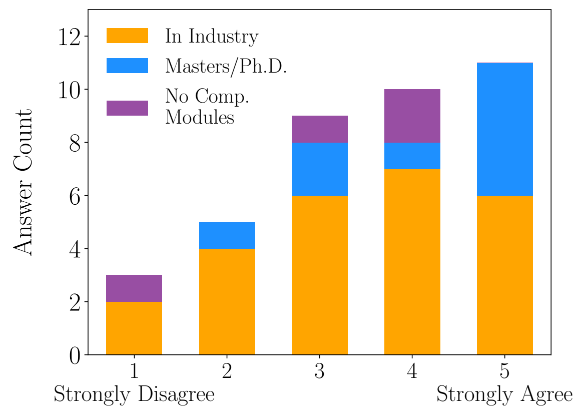
Figures 4(e,f) tell a different story, but not necessarily a contradicting one. Alumni in Ph.D./Masters are less confident in applying to computation related jobs, additionally they also think that they lost more opportunities compared to their peers in industry. It seems that the advantages of computational skills are more significant for Ph.D. and Master's students, and at the same time they tend to suffer more from the lack of these skills. A student who will pursue a Ph.D./Masters degree needs and benefits more from stronger computational knowledge. While the core curriculum should provide enough computational material to satisfy the needs of industry careers, options should be made available to cover the extra needs of students that may choose a Ph.D./Masters path. It is also worth noting that individuals pursuing Ph.D./Masters may tend to value the courses they take more, regardless of what their job is and this could partially be a reason for the difference between the two groups.

Satisfaction with the computational curriculum

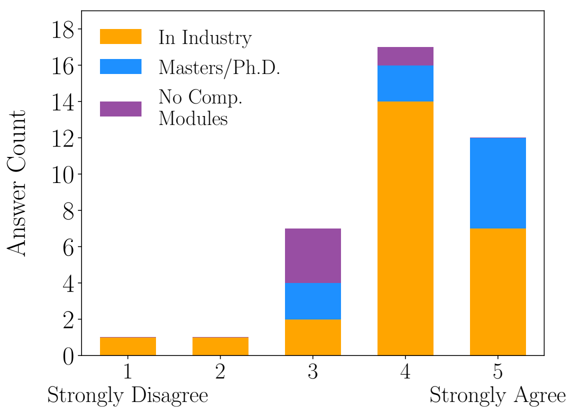
To measure the satisfaction of the graduates with the quantity of computational material in the curriculum, the following question was asked in the survey: *"Should there be more or less computational materials in the MatSE curriculum?"* Almost 50% of the respondents have neutral opinion on the 5 point Likert scale (Figure 5a), indicating that they think the amount of computational material they received is close to ideal. Still, on average there seems to be a moderate demand for more computational material, mostly originating from the Ph.D./Masters students. The average opinion on the importance of computational skills in MSE and in an alumni's career are both strongly positive (Figure 5(b,c)). However, as pointed out in the "Impact on Career" section, the current benefits of the computational curriculum of MatSE is not reported to be as strong (Figures 4(a,b,c)). This could indicate that the amount of potential benefits of computational skills is larger than what our graduates are currently experiencing. Therefore, there is room for qualitative improvement on the computational material than quantitative, since they are satisfied with the amount of material.



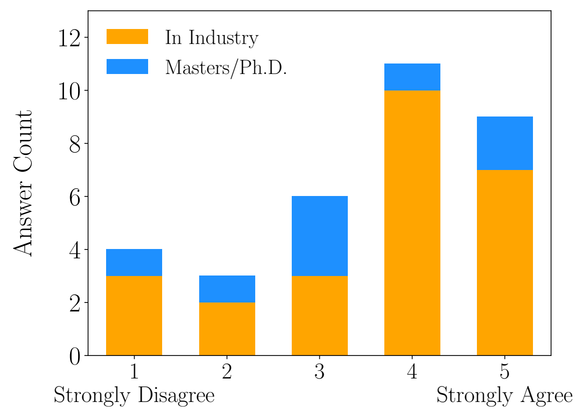
(a) Should there be more or less computational material in MatSE classes?



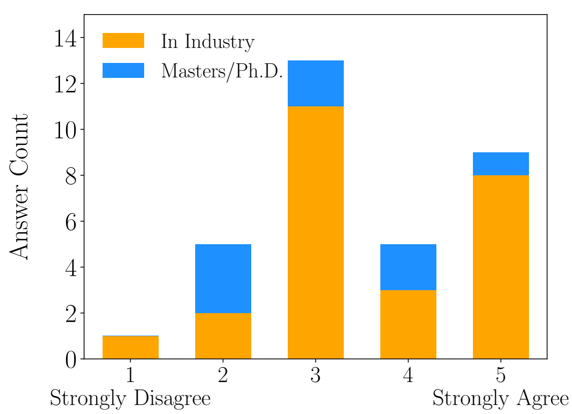
(b) Do you think computational tools and skills are important for your career?



(c) Do you think computational tools and skills are important for materials science and engineering?



(d) MatSE computational curriculum has prepared you to learn new computational skills by yourself.



(e) MatSE computational curriculum should focus more on general computational thinking skills rather than introducing tools specific to materials science.

Figure 5: Broad Impact

Conclusions

Among the three classes of computational skills (simulation tools, programming, and data analysis), data analysis is the most relevant one for the graduates. Python is by far the most common programming language used by the graduates, therefore our students may benefit from more computational content that combine data analysis and Python. Apart from FEA, simulation tools are rarely used by our alumni in industry. Thus, the computational curriculum has a more profound impact on Ph.D./Masters students compared to alumni in industry. Ph.D./Masters students benefit more from the computational content and report more negative consequences when lacking computational skills. An ideal MSE computational curriculum should be sufficiently flexible to respond to the distinct needs of these two career paths. The benefit of computational material in the MSE curriculum manifests itself after getting the job and it does not help much during the job search. Coding related skills such as testing and verifying code or working with complicated chunks of code is not critically important for the alumni, but it would be useful monitor these trends for the future needs of our alumni. On average, our graduates find the amount of computational content to be ideal, but there is still room for qualitative improvement, since the respondents think that computational skills could potentially have been more beneficial to their career than they currently are. Given the satisfaction with the amount of computational content in our curriculum, we think that our computational program can serve as a good starting point for other materials science programs. Readers can learn about the details of the computational curriculum from our previous publications [7, 8, 9, 10]. In light of the survey results, our aim is to modify the curriculum to focus more on data analysis with Python and finite element methods since these are the most relevant computational skills for graduates. We believe others can also benefit from following this direction.

The computational content of our program was not reminded to the survey respondents. We relied on them remembering their experience with the modules. This should be considered as a limitation together with the small sample size (n=38) while interpreting the findings of our study. Future studies with higher number of participants will be needed for more solid conclusions.

Acknowledgement

This work received IRB approval under protocol number 23547 and was supported by the College of Engineering and the Department of Materials Science and Engineering at the the University of Illinois Urbana-Champaign as part of the Strategic Instructional Initiatives Program (SIIP). The authors thank Prof. Matthew West and Prof. Luke Olson for helpful discussions.

References

[1] K. Thornton, S. Nola, R. Edwin Garcia, M. Asta, and G. B. Olson, "Computational materials science and engineering education: A survey of trends and needs," *JOM*, vol. 61, pp. 12–17, 2009.

[2] R. A. Enrique, M. Asta, and K. Thornton, "Computational materials science and engineering education: An updated survey of trends and needs," *Jom*, vol. 70, pp. 1644–1651, 2018.

[3] R. Mansbach, A. Ferguson, K. Kilian, J. Krogstad, C. Leal, A. Schleife, D. Trinkle, M. West, and G. Herman, "Reforming an undergraduate materials science curriculum with computational modules," *Journal of Materials Education*, vol. 38, no. 3-4, pp. 161–174, 2016.

[4] R. A. Mansbach, G. L. Herman, M. West, D. R. Trinkle, A. Ferguson, and A. Schleife, "Work in progress: Computational modules for the matse undergraduate curriculum," in *2016 ASEE Annual Conference & Exposition*, 2016.

[5] A. Kononov, P. Bellon, T. Bretl, A. L. Ferguson, G. L. Herman, K. A. Kilian, J. A. Krogstad, C. Leal, R. Maass, A. Schleife *et al.*, "Computational curriculum for matse undergraduates," in *2017 ASEE Annual Conference & Exposition*, 2017.

[6] K. Kang, M. D. Goodman, J. A. Krogstad, C. Leal, P. Y. Huang, and A. Schleife, "Incorporating the use of a materials database into a materials science and engineering freshman course," in *ASEE annual conference*, 2021.

[7] Y. Dan, A. Schleife, D. Trinkle, P. Huang, and C. Leal, "Impact of transitions between online and offline learning during covid-19 on computational curricular reform: Student perspective," in *2022 ASEE Annual Conference & Exposition*, 2022.

[8] G. Lu, D. Trinkle, A. Schleife, C. Leal, J. Krogstad, R. Maass, P. Bellon, P. Huang, N. Perry, M. West *et al.*, "Impact of integrating computation into undergraduate curriculum: New modules and long-term trends," in *ASEE Virtual Annual Conference*, 2020.

[9] X. Zhang, "Computational curriculum for matse undergraduates and the influence on senior classes," in *ASEE Annual Conference proceedings*, 2018.

[10] C.-W. Lee, A. Schleife, D. R. Trinkle, J. A. Krogstad, R. Maass, P. Bellon, J. K. Shang, C. Leal, M. West, T. Bretl *et al.*, "Impact of computational curricular reform on non-participating undergraduate courses: Student and faculty perspective," in *2019 ASEE Annual Conference & Exposition*, 2019.

[11] S.-H. Hsu, Y.-C. Wang, C.-J. Cheng, and Y.-F. Chen, "Developing a decomposed alumni satisfaction model for higher education institutions," *Total Quality Management & Business Excellence*, vol. 27, no. 9-10, pp. 979–996, 2016.

[12] R. J. Puerzer and D. M. Rooney, "The alumni survey as an effective assessment tool for small engineering programs," *Journal of Engineering Education*, vol. 91, no. 1, pp. 109–116, 2002.

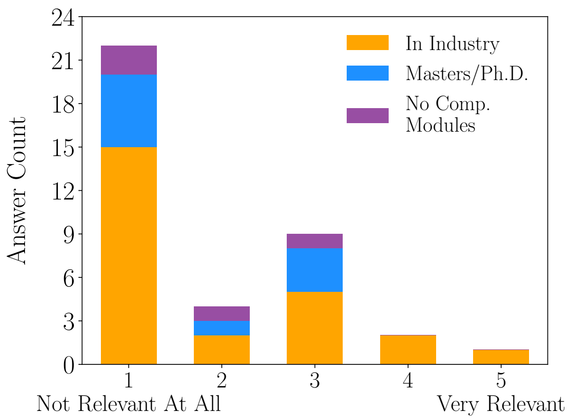
[13] A. F. Cabrera, D. J. Weerts, and B. J. Zulick, "Making an impact with alumni surveys," *New Directions for Institutional Research*, vol. 2005, no. 126, pp. 5–17, 2005.

[14] J. F. Volkwein, "Assessing alumni outcomes," *New Directions for Institutional Research*, vol. 2010, no. S1, pp. 125–139, 2010.

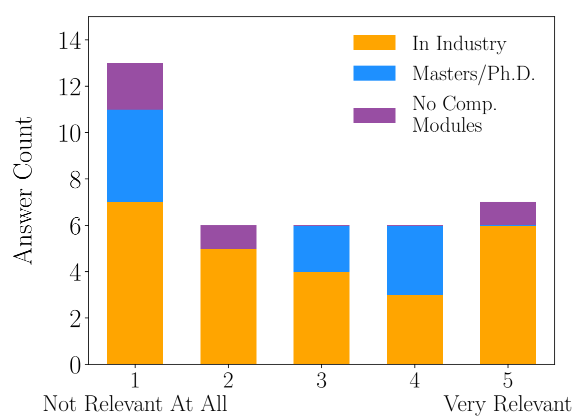
[15] J.-O. Andersson, T. Helander, L. Höglund, P. Shi, and B. Sundman, "Thermo-calc & dictra, computational tools for materials science," *Calphad*, vol. 26, no. 2, pp. 273–312, 2002.

[16] H. H. Cheng, "C for the course," *Mechanical Engineering*, vol. 131, no. 9, p. 50, 2009.

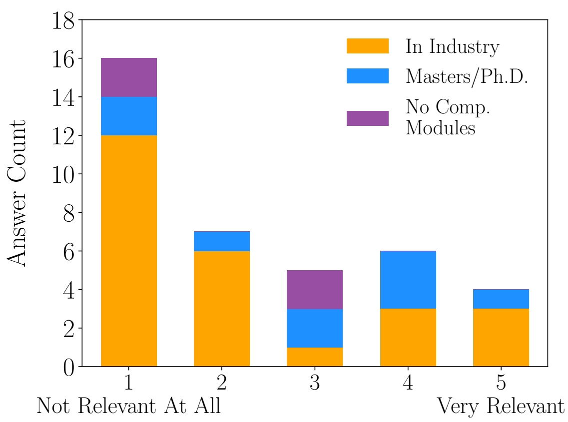
Supplementary Information



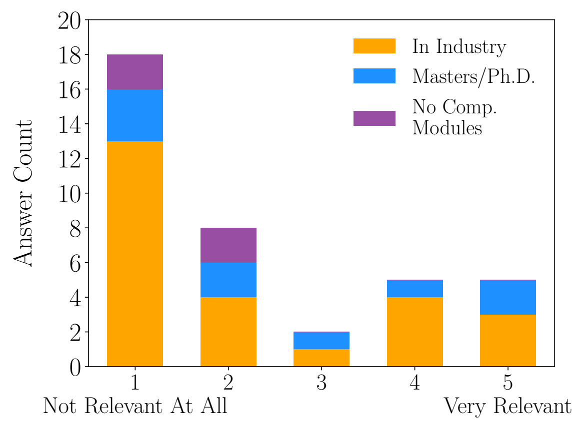
(a) Working with big projects collaboratively



(b) Managing workflows and version control



(c) Testing and verifying code



(d) Finding your way around complicated chunks of code

Figure S1: Coding related skills

All of the questions asked in the survey are listed below:

- Q1: “What is your year of graduation?”
- Q2: “Have you pursued any other degrees after graduation?”
- Q3: “How long is your industry experience in total, since graduation?”
- Q4: “What is your job title?”
- Q5: “How long is your experience in your current field?”
- Q6: “Which of the following best describes the level of your current position?” (Entry, Mid, Senior, Executive)
- Q7: “What is the size of your current employer company?” (Micro-sized (less than 10 employees), Small-sized (10-100 employees), Medium-sized (100-1000 employees), Large sized (more than 1000 employees))

- Q8: “Which sector does your current employer operate in?”
- Q9: “Did you have any computational material in your MatSE curriculum?” (Yes,No)
- Q10: “The computational part of the MatSE curriculum helped me getting the job I wanted.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q11: “The computational part of the MatSE curriculum helped me performing better at my job.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q12: “The computational part of the MatSE curriculum has expanded my career opportunities.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q13: “The computational part of the MatSE curriculum has an impact on my career choices.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q14: “I could confidently apply for jobs requiring computational skills when I graduated.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q15: “I think I’ve missed several opportunities due to lack of computational skills.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q16: “How often do you use computational/simulation tools for your job?” (Never - ... - Very Often, 5 point Likert Scale)
- Q17: “Which tools/methods are you using? Please check all that apply.” (Finite Element Analysis, Molecular Dynamics, Density Functional Theory, CALPHAD - Computer Coupling of Phase Diagrams and Thermochemistry, No, “I’m not using any computational tools.”, Other)
- Q18: “How often are you working with programming languages for your job?” (Never - ... - Very Often, 5 point Likert Scale)
- Q19: “Which programming languages are you working with? Please check all that apply.” (Python, Matlab, C++/C, Java/Javascript, “I’m not working with any programming languages”, Other)
- Q20: “How often do you perform data analysis for your job?” (Never - ... - Very Often, 5 point Likert Scale)
- Q21: “Which tools/languages are you using for data analysis? Please check all that apply.” (Excel, Python, Matlab, R, System Applications and Products in Data Processing (SAP), “I’m not performing any data analysis”, Other)
- Q22: “MatSE computational curriculum has prepared you to learn new computational skills by yourself.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q23: “MatSE computational curriculum should focus more on general computational thinking skills rather than introducing tools specific to materials science.” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q24: “How relevant ’*Working with big coding projects collaboratively*’ is for your job?” (Not relevant at all - ... - Very relevant, 5 point Likert Scale)
- Q25: “How relevant ’*Managing workflows and version control*’ is for your job?” (Not relevant at all - ... - Very relevant, 5 point Likert Scale)
- Q26: “How relevant ’*Testing and verifying code*’ is for your job?” (Not relevant at all - ... - Very relevant, 5 point Likert Scale)
- Q27: “How relevant ’*Finding your way around complicated chunks of code*’ is for your job?” (Not relevant at all - ... - Very relevant, 5 point Likert Scale)
- Q28: “Do you think computational tools and skills are important for materials science and engineering?” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)
- Q29: “Do you think computational tools and skills are important for your career?” (Strongly Disagree - ... - Strongly Agree, 5 point Likert Scale)

- Q30: “Should there be more or less computational material in MatSE classes?” (Less - ... - More, 5 point Likert Scale)
- Q31: “What are the topics that are essentially effective for your career in computational MatSE curriculum?”
- Q32: “What are the topics that were missing for your career in computational MatSE curriculum?”