



A Framework for Integrating Computational Simulations into Engineering Lessons

Dr. Kerrie Anna Douglas, Purdue University, West Lafayette

Dr. Douglas is a Visiting Assistant Professor in the Purdue School of Engineering Education. Her research is focused on methods of assessment and evaluation unique to engineering learning contexts.

Dr. Tanya Faltens, Purdue University, West Lafayette

Tanya Faltens is the Educational Content Creation Manager for the Network for Computational Nanotechnology (NCN) which created the open access nanoHUB.org cyber-platform. Her technical background is in Materials Science and Engineering (Ph.D. UCLA 2002), and she has several years' experience in hands-on informal science education, including working at the Lawrence Hall of Science at UC Berkeley. While at Cal Poly Pomona, she taught the first year engineering course, mentored student capstone research projects, and introduced nanoHUB simulation tools into the undergraduate curriculum in materials science and engineering and electrical engineering courses. Much of her work has focused on introducing STEM concepts to broad audiences and encouraging students, including women and others in traditionally under-represented groups, to consider graduate school. Four of her former research students are currently in, or have completed, Ph.D. programs and a few more are in the pipeline.

Prof. Heidi A. Diefes-Dux, Purdue University, West Lafayette

Heidi A. Diefes-Dux is a Professor in the School of Engineering Education at Purdue University. She received her B.S. and M.S. in Food Science from Cornell University and her Ph.D. in Food Process Engineering from the Department of Agricultural and Biological Engineering at Purdue University. She is a member of Purdue's Teaching Academy. Since 1999, she has been a faculty member within the First-Year Engineering Program, teaching and guiding the design of one of the required first-year engineering courses that engages students in open-ended problem solving and design. Her research focuses on the development, implementation, and assessment of modeling and design activities with authentic engineering contexts. She is currently a member of the educational team for the Network for Computational Nanotechnology (NCN).

Krishna Madhavan, Purdue University, West Lafayette

Dr. Krishna Madhavan is an Assistant Professor in the School of Engineering Education at Purdue University. He is Co-PI and Education Director of the NSF-funded Network for Computational Nanotechnology (nanoHUB.org which serves over 330,000 global researchers and learners annually). Dr. Madhavan was the Chair of the IEEE/ACM Supercomputing Education Program 2006. In January 2008, he was awarded the US National Science Foundation (NSF) CAREER award for work on learner-centric, adaptive cyber-tools and cyber-environments. He was one of 49 faculty members selected as the nation's top engineering educators and researchers by the US National Academy of Engineering to the Frontiers in Engineering Education symposium. Dr. Madhavan leads a major NSF funded effort called Deep Insights Anytime, Anywhere (DIA2) that attempts to characterize the impact of NSF and other federal investments in the area of science, technology, engineering, and mathematics education using interactive knowledge mining and visual analytics for non-experts in data mining. DIA2 is currently deployed inside the NSF and is already starting to affect federal funding policy. Dr. Madhavan also served as Visiting Research Scientist at Microsoft Research, Internet Services Research Group. His research has been published in Nature Nanotechnology, IEEE Transactions on Computer Graphics and Applications, IEEE Transactions on Learning Technologies, and several other top peer-reviewed venues. Dr. Madhavan currently serves as PI or Co-PI on federal and industry funded projects totaling over \$20M.

A Framework for Integrating nanoHUB.org Computational Simulation Tools into Engineering Lessons

Abstract

Nanotechnology related education is a critical part of a multidisciplinary engineering education. While nanoHUB.org simulations are readily available for instructors to use in their courses as supplemental course material for teaching nanotechnology concepts, maximum student learning through the use of simulations can only be realized through the careful alignment of student learning needs and the value added by inserting simulation use into a course. Computational simulations are unique in that the developed tools are adopted for a purpose, *learning*, other than what they were originally designed for, *research*. The approach instructors take to integrate simulations will have a direct impact on what students learn from the simulation activity. Therefore, based on our experiences with computational simulation in the classroom, we extend practices in curriculum development to propose an integration method that will assist faculty in effectively incorporating research-grade simulations into their existing courses.

Introduction

Nanotechnology is a rapidly growing area of research¹, therefore training students to understand atomic level principles and have the skills necessary to work in nanotechnology-related projects is critical in all fields of engineering and part of a multidisciplinary engineering education². Yet, the growing body of research related to nanotechnology education has consistently found that students struggle with concepts at the nano-scale³⁻⁶. Simulation tools developed for research can provide deep insight into physical processes, dramatic interactive visualizations of phenomena that might be otherwise invisible, and the ability to explore *in silico*, as one might explore in a lab, but on a much shorter time frame. Increased access to significant computing power and efficient simulation code has brought powerful computational simulation tools within the reach of engineering researchers, instructors, and students worldwide, making visual the atomic level processes. Instructors have been utilizing their access to computational simulations and incorporating them into courses in a variety of ways, with the intuitive understanding that students learn from seeing what occurs at the nanoscale and being able to manipulate variables at that level⁷. While computational simulation is a powerful tool for research and can also be applied to education, care must be taken to effectively transfer tools from the research realm to the educational realm to ensure desired student learning outcomes.

nanoHUB.org, an open access science gateway created by the Network for Computational Nanotechnology (NCN), provides an online mechanism for creating and sharing computational simulation tools that need no installation and are run on nanoHUB's cloud computing resources⁸. To date, over 350 simulation tools from around the world are hosted on nanoHUB. There are 44 simulation tools that are NCN supported for educational purposes. Due to the large number of users and availability of computational simulations to anyone with Internet access, we focus our discussion on the computational simulation tools available through nanoHUB.org in this paper.

The ability to virtually explore physical phenomena enables curricular activities to support a range of learning levels, specified by Bloom's Revised Taxonomy of Educational Objectives³.

Instructors are able to assign computational simulation activities that challenge students at the desired level of learning. Students are able to not only see phenomena, but to introduce new variables and manipulate the phenomena through experiments and then make evaluations, which tap higher levels of learning. One simulation example of this is the Crystal Structures – Packing Efficiency¹⁰. Supportive of the highest level of Bloom’s level of learning, computational simulations can enable students and researchers to apply information about nano-scale phenomena to create mathematical models to explain and predict behaviors at the nano-scale.

Despite the range of educational uses, computational simulation tools do not generally come embedded within a curriculum. While nanoHUB.org offers a space for contributors to share their educational resources, these must be adopted for an instructors’ specific purpose. Instructors use computational simulations as part of courses in a variety of ways⁷. For example, while one instructor might use a simulation lab to teach students about plastic deformation[#], other instructors might use the same computational simulation to teach students about the role of simulation in engineering research. Taking an existing tool used for research and bringing it into the classroom as part of the curriculum is very different starting point than creating technology specifically for classroom use. Yet, both require evidence and rationale to justify intended use. Computational simulations were designed for real-world research, and adopting them requires thought about one’s goals and design of curriculum to achieve those goals. There are many potential uses for computational simulations, yet to maximize student learning, a theoretically based framework is needed to assist instructors in the integration of simulations in the classroom. This work presents the nanoHUB framework of computational simulation pedagogy, based on cognitive learning theory, *How People Learn* (HPL)¹¹, to provide practical guidelines for engineering instructors in integrating computational simulations into their courses in order to effectively support student learning.

Educational Simulations Versus Computational Simulations

In an educational context, the term *simulation* has been broadly used to describe a “computer program in which it [the simulation] temporarily creates a set of things through the means of a program and then relates them together through cause and effect relationships,”⁶. We will use the term educational simulation to refer to those that are typically created to address student conceptual understanding, and that are packaged along with other curricular materials such as lesson plans and learning modules. Educational simulations typically have animation and vary considerably in terms of activities, from *serious games*¹² designed to mimic real life scenarios to virtual physics labs, such as ThinkerTools¹³. Educational simulations are designed and intended for one purpose, that is, student learning. They are not intended for a research context; they do not generate new knowledge, rather they enable learning of previously discovered knowledge. Educational simulations are analogous to a calculator; students input values and an output is generated. A calculator does not teach students how to multiply, but it will give students the answer to a multiplication question. In the same way, educational simulations tell students the answers, but are quite different from the computational simulation tools used by researchers to solve real world science and engineering problems.

Magana, Brophy, and Bodner⁷ define *computational simulations*, as “working representations of reality that are used in training, research, and education to represent physical phenomena,

devices, and/or processes based on mathematical models and numerical solution techniques executed on supercomputers or distributed-computing platforms,” (p. 101). These tools were developed to make contributions to the advancement of research in engineering and science, by allowing researchers to know how the simulation works and being transparent about the process. Computational simulations are like a glass box; they enable the user to not only input variables, but to see how the output is generated. In other words, one of the distinguishing features of a computation simulation is that the simulation shows how the answer was derived.

Computational simulations available on nanoHUB.org have been used in the classroom to teach students about the content the simulations were designed to model^{4,6,7,9}. Magana and colleagues⁷ identified four characteristics of the simulation tools offered in nanoHUB.org that make them particularly useful for instructional purposes. Computational simulations are: (1) domain specific; (2) easily accessible; (3) consistent in terms of graphical user interface, and (4) authentic research tools. Furthermore, Magana and colleagues conducted a qualitative study of instructors who use nanoHUB.org computational simulations in the classroom and found that they used simulations for eight different learning purposes, which can be categorized as serving two general purposes: learning through *building simulations* and learning through *using simulations*.

Integration of *How People Learn* with Classroom Use of Computational Simulations

While instructors may have differing purposes and ways of integrating computational simulations in their lessons, the synthesis of research from the learning sciences can inform what promises to be most effective for student learning. The process of creating good instructional materials often follows traditional instructional design methods. For example, Smith and Ragan¹⁴ instructional design methods begin with an analysis of the learning environment, the learner, and the learning tasks. Curricular and instructional materials are then designed to directly address the learning need, and evaluation is included throughout the curriculum development process. Computational simulations, by contrast, emerge from a completely different process. Subject matter experts design computational simulations to address their *research* needs, with no consideration of a learner or learning tasks. Since many instructors utilize computational simulation tools for educational purposes, a pragmatic pedagogical approach is needed to assist in the integration of simulations, based on best practices in education, to ensure that the simulations are utilized effectively as part of a well-designed curriculum.

To answer the question, “How can computational simulations be effectively used as a pedagogical tool?” we draw from Bransford’s framework of How People Learn (HPL)¹¹, based on a synthesis of and our experiences with a sophomore materials science course. The NRC report, *How People Learn (HPL)*¹¹, synthesized the research literature concerning the ways that new information is learned and conceptual change occurs in a format that is easily digestible for a wide audience. In addition, Wankat¹⁵ discussed direct HPL applications for engineering faculty. From the NRC report and Wankat’s discussion, we have noted foundational tenets of *HPL* and provide an example of how student learning could be supported when using computational simulation in the classroom^{5,14}.

Foundational Tenets of the How People Learn Theory

Knowledge is actively constructed based on preconceptions, a concept referred to as constructivism. Misconceptions can develop as students try to assimilate new information with their previous understanding. For example, understanding how materials behave differently at the atomic level is a challenge to students who have learned material behavior at other scales¹⁷.

Because students are constructing their knowledge through their prior understandings, misconceptions must be directly confronted through activities that dispute the misconceptions. One way to confront misconceptions related to atomic-level behavior is to give students a pre-lab assessment to record their prior knowledge on paper, then have them conduct the simulation lab and note whether their prior knowledge was correct, and then discuss the results and their thinking processes as a group. For example, the pre-lab could ask students to report what they think will happen to a sample when a new stimulus is introduced. Students then perform the simulation experiment to see what actually happens. In the lab report, students discuss both what they thought would happen and what actually happened. Instructors can additionally lead a classroom discussion to process any of their misconceptions about the expected material behavior and their observations of what actually happened in the simulation. By clearly articulating and capturing on paper what their initial concepts or misconceptions are, and then discussing the actual results in comparison with their initial ideas, students' are more likely to recognize any previous misconceptions as incorrect understandings.

Students need to be engaged in and monitor their own learning process, referred to as metacognition. Pre-lab and lab reports provide an opportunity to encourage student reflection on their own learning. A section where this reflection is explicitly required can easily be included in an existing lab report assignment.

Students need opportunities to practice new skills and to receive feedback directly related to these attempts. Because nanoHUB provides online access to simulation, instructors can have students perform multiple simulations with different scenarios, so that students have multiple exposures to the content with feedback from the simulation and/or the instructional team.

Experts conceptualize how specific content is related at a broader understanding level, whereas novices tend to focus on specific content without conceptualizing the broader importance. Embedding content learned directly into assigned designed projects provides students the opportunity to see how the specific content learned has broader implications. For example, instructors could assign a design project in which students must run a simulation to generate data for their design work.

nanoHUB Framework for Using Computational Simulations to Enhance Student Learning

Being informed by the *HPL* framework¹¹, Wankat's¹⁵ discussion of *HPL* applications to engineering learning, curriculum design principles, and experiences[#] working with instructors who use computational simulation in the classroom, we recommend the following steps as a pedagogical framework to assist instructors in the integration of computational simulation for learning. These steps are for the instructor who has decided to use a specific computational simulation tool in a course

- 1. Identify difficult concepts that can be addressed by the simulation.** It is important to relate a simulation back to students' learning needs. Instructors are often aware from previous experience what concepts students struggle with. In addition, formal analysis of student assignments and exam responses can reveal the prevalence of different misconceptions around each concept.
- 2. Create one or two key learning goals that can be accomplished through use of computational simulation.** Learning goals are the “big ideas” that instructors want students to understand. With the variety of affordances provided by a computational simulation, it is tempting for instructors to identify many learning goals. When there are too many learning goals, however, students may not learn any one concept very well. Sometimes this is referred to as “content tyranny” or trying to cover too many topics. To be most effective, it is helpful for instructors to keep in mind the primary value added to student learning by the simulation.
- 3. Develop specific learning objectives to be achieved with the simulation.** Whereas learning goals are the broad concepts, learning objectives are specific expectations of what students should be able to know and do. A good rule of thumb is to articulate two to four learning objectives for each learning goal. The objectives are the specific, measurable ways that instructors will know the learning goal has been achieved. Each learning objective should specify the content and the depth of learning that instructors expect students to achieve. Kubiszyn and Borich¹⁶ provide guidelines on the creation of learning objectives. There are three components to an effective learning objective: an observable behavior, the conditions under which the behavior is displayed, and the level of achievement expected. The observable behavior in a learning objective is to be aligned with level of learning instructors expect their students to achieve.
- 4. Create or refine curricular materials to align with the learning objectives.** The learning objectives inform the specific content to be covered in the curricular materials that are used in conjunction with the simulation. For example, in situations where computational simulations are used as a lab experiment, the prelab questions, instructions for how to complete lab, lab report, and group discussion questions should all be directly mapped to the learning objectives. The prelab questions can be an opportunity for students to state their misconceptions. Then, in group or classroom discussion, students can revisit their previous misconceptions as a way of reinforcing their new learning. In addition, when instructors provide clear guidance on the specific learning objectives, students are able to identify what they should be learning through the simulation. Students also benefit from opportunities to process learning with peers and their instructors.
- 5. Map simulation activities to the learning objectives.** There are many ways that instructors could use simulations. The learning objectives inform the activities that instructors assign students to complete in the simulation. As instructors are considering the activities within the simulation for students, one question that might be considered is, *What simulation activities will students do that specifically address their misconceptions or build deeper conceptual learning?* Another question is, *What specifically will students do in the simulation that directly supports their learning the objectives identified as most important?*

- 6. Assess student learning and provide ongoing feedback.** When students receive direct and timely feedback, they can learn from their mistakes. Computational simulations can be designed to provide students immediate feedback on their work. For example, students are able to conduct a lab and see the results. The lab report questions, quizzes, and exams related to these concepts are created from the learning objectives. Each objective should be represented in the lab report, quizzes, and exams at the same level of Bloom's taxonomy as specified in the learning objective. In addition to providing individual student feedback, instructors can assess student learning as a whole and consider whether there are areas students continue to struggle with or whether students have shown proficiency.
- 7. Evaluate and reflect on the integration.** Reflecting on the integration of simulation can help inform improvement of integration of the simulation in future semesters. Especially in the first two or three years, conducting formative evaluation through student feedback, student work, and discussion with the instructional team can provide insight into what went well and what could be modified. Some of the questions that might be addressed are: *Did students know what to do to run the simulation? Did students explicitly know (from the lab instructions) what content they were to learn by using the simulation? Did students demonstrate their understanding of the concept(s)? In what ways did students not understand the concept(s)? How can the alignment between students' understanding of learning expectations, your expectations, and assessment be clearer?*
- 8. Refine course materials and simulation.** Instructors can use the information collected and discovered through evaluation and reflection to identify what areas of the curriculum and computational simulation could be enhanced or modified.

Figure 1 shows our Framework for Integration of Computational Simulation in the classroom. Red X's represent difficult concepts that are identified by analyzing student responses and looking for common student misconceptions and aspects that seem difficult. Green circles represent the one or two larger learning goals for the lesson. From the learning goals, a few learning objectives (A, B, C), represented as blue squares, are developed. The learning objectives are mapped to different levels of Bloom's taxonomy. Depending on the complexity of the simulation tool, learning objectives up to the "Analyze" and "Synthesize" levels can be achieved. In the figure, the three learning objectives A, B, and C map to the levels "Apply" and "Understand".

Activities are created to help the student achieve each learning objective, and are represented as yellow triangles. Again, the letters A, B, and C are used to show the mapping of specific activities with their corresponding learning objectives and their levels of Bloom's taxonomy. These activities can be simulation-based activities as well as other curricular activities. After students engage with the activities, the learning objectives are assessed. The assessment results are represented as green pluses and a red minus, representing students doing well on the assessment or not. The instructional team reflects on the results of the activity and assessment and can then make revisions to the curricular material to increase student achievement of the learning goals in subsequent implementations.

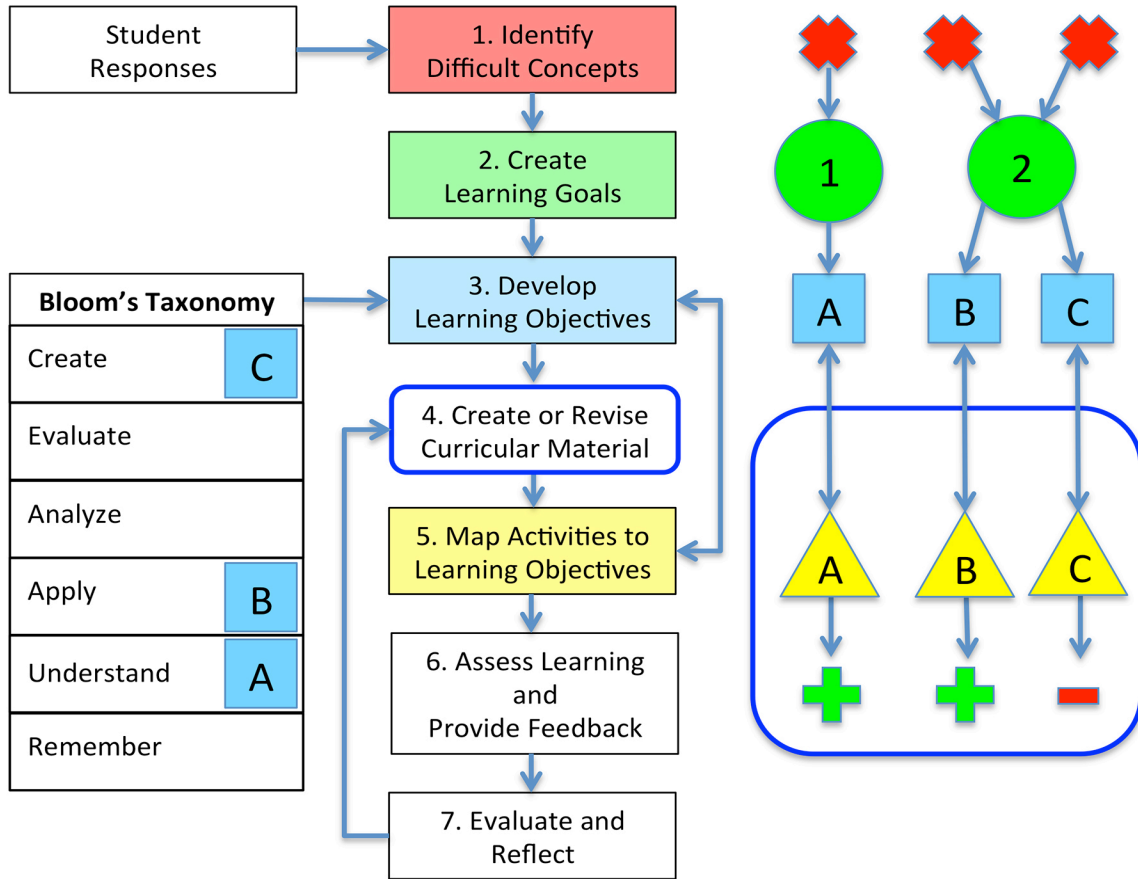


Figure 1. nanoHUB's Framework for Integration of Computational Simulation.

Conclusion and Implications

Nanotechnology is multidisciplinary and has implications for all fields of engineering. Computational simulations are important tools that engineers use to advance nanotechnology. Computational simulations developed for research purposes are migrating into classroom use where there is great potential for meaningful integration that will support student learning of difficult concepts. However, maximum student learning through the use of simulations can only be realized by applying sound instructional practices to align what students are expected to learn with the value added by inserting a research-based simulation use into a course.

To prepare engineering students for the rapidly growing field of nanotechnology, there is a need for pedagogical frameworks, based on learning sciences research, to practically assist instructors to use computational simulation in their classrooms. Arguably, the approach instructors take to integrate simulations will have a direct impact on what students learn from the simulation activity. We propose a pedagogical framework for the integration of computational simulation that will assist faculty in effectively incorporating computational simulations into their existing courses. Instructors who are interested in integrating simulations into their classroom have a starting point of how to design curricular materials and use the simulation in a complimentary manner, in a way that is aligned with research findings from the learning sciences.

The nanoHUB framework for integrating computational simulation is based on a synthesis of research from the learning sciences¹¹ and our experiences working with instructors who have used computational simulation in their classrooms. As such, this framework is developed as a resource for instructors interested in using computational simulation in their classroom. Future research should empirically evaluate the nanoHUB framework for the integration of computational simulation to support student learning of nanotechnology related concepts.

Bibliography

1. Roco, M. C., Mirkin, C. A. & Hersam, M. C. Nanotechnology research directions for societal needs in 2020: summary of international study. *J. Nanoparticle Res.* **13**, 897–919 (2011).
2. Fonash, S. J. Education and training of the nanotechnology workforce. *J. Nanoparticle Res.* **3**, 79–82 (2001).
3. Light, G. *et al.* Understanding undergraduate students' conceptions of a core nanoscience concept: Size and scale. in *Proceedings of the First International Conference on Research in Engineering Education* (2007). at <<http://nanotechnology.wmwikis.net/file/view/Light+202007+Paper.pdf/265469902/Light+202007+Paper.pdf>>
4. Beophy, S. P., Magana, A. J. & Strachan, A. Lectures and Simulation Laboratories to improve Learners' Conceptual Understanding. *Adv. Eng. Educ.* **3**, 1–27 (2013).
5. Lu, K. A Study of Engineering Freshmen Regarding Nanotechnology Understanding. *J. STEM Educ. Innov. Res.* **10**, 7–16 (2009).
6. Coughlan, A. *et al.* Enhanced Learning of Mechanical Behavior of Materials via Combined Experiments and nanoHUB Simulations: Learning Modules for Sophomore MSE Students. in *MRS Proceedings* **1762**, mrsf14–1762 (Cambridge Univ Press, 2015).
7. Magana, A. J., Brophy, S. P. & Bodner, G. M. Instructors' Intended Learning Outcomes for Using Computational Simulations as Learning Tools. *J. Eng. Educ.* **101**, 220–243 (2012).
8. Madhavan, K., Zentner, M. & Klimeck, G. Learning and research in the cloud. *Nat. Nanotechnol.* **8**, 786–789 (2013).
9. Klimeck, G., McLennan, M., Brophy, S. P., Adams, G. B. & LUNDSTROM, M. S. nanoHUB.org: Advancing Education and Research in Nanotechnology. *Comput. Sci. Eng.* **10**, 17–23 (2008).
10. Vasileska, D. & Klimeck, G. Crystal Structures - Packing Efficiency Exercise. (2010). at <<https://nanohub.org/resources/9154>>
11. Bransford, J. D., Brown, A. L. & Cocking, R. R. *How people learn: Brain, mind, experience, and school.* (National Academy Press, 1999). at <<http://doi.apa.org/psycinfo/1999-02807-000>>
12. Michael, D. R. & Chen, S. L. *Serious games: Games that educate, train, and inform.* (Musk & Lipman/Premier-Trade, 2005). at <<http://dl.acm.org/citation.cfm?id=1051239>>
13. White, B. Y. & Frederiksen, J. R. Inquiry, Modeling, and Metacognition: Making Science Accessible to All Students. *Cogn. Instr.* **16**, 3–118 (1998).
14. Smith, P. L. & Ragan, T. J. *Instructional design.* (Wiley New York, NY, 1999)
15. Wankat, P. C. Improving engineering and technology education by applying what is known about how people learn. *J. SMET Educ. Innov. Res.* **3**, 3–8 (2002).
16. Kubiszyn, T. & Borich, G. *Educational testing and measurement.* (Harper Collins Publishers, 2007).
17. Coughlan, A., Johnson, D., Diefes-Dux, H.A., Douglas, K.A., Erk, K., Faltens, T.A., Strachan, A. Enhanced learning of mechanical behavior of materials via combined experiments and nanoHUB simulations: Learning modules for sophomore MSE students. *Proceedings of the Materials Research Society Symposium, Boston, MA, 1762, mrsf14-1762-aaa06-03 doi:10.1557/opl.2015.152* (2014).