Computational Molecular Science and Engineering Forum
for the combined community of engineers and scientists developing and applying molecularly based theories, modeling, and simulation
http://comsef.org/

Newsletter

AIChE

October 2025

In This Issue

Vashisth Winner of 2025 CoMSEF Impact Award

Professor Harish Vashisth, the John T. and Doris H. Carpenter Professor of Chemical Engineering & Bioengineering at the University of New Hampshire, is the winner of the 2025 CoMSEF Impact Award. He is cited "***for outstanding contributions to advancing molecular simulation methods for mechanistic studies of biophysical and self-assembly phenomena in proteins, RNA, and colloids***." Prior to joining the faculty at UNH in 2013, Harish completed a postdoctoral research fellowship with Prof. Charles L. Brooks III at the University of Michigan (2010-13) and a PhD in chemical and biological engineering with Cameron F. Abrams at Drexel University (2005-10). Harish will deliver a presentation describing his research during the CoMSEF Plenary Session at the 2025 AIChE Annual Meeting. The CoMSEF Impact Award recognizes outstanding research in computational molecular science and engineering, encompassing both methods and applications, and is given annually to a CoMSEF member who is within 15 years of completion of their highest degree.

Coley Winner of the 2025 CoMSEF Young Investigator Award

Professor Connor Coley, Associate Professor of Chemical Engineering at the Massachusetts Institute of Technology (MIT) is the 2025 winner of the CoMSEF Young Investigator Award. He is cited "***for his extensive contributions to molecular design, synthesis planning, and structure elucidation with modeling frameworks employing modern computational and deep learning techniques***." Prior to joining the MIT faculty in 2020, Connor completed a postdoctoral fellowship and PhD in chemical engineering also at MIT. Connor will deliver a presentation describing his research during the CoMSEF Plenary Session at the 2025 AIChE Annual Meeting. The CoMSEF Young Investigator Award recognizes outstanding research in computational molecular science and engineering, encompassing both methods and applications, and is given annually to a CoMSEF member who is within 7 years of completion of their highest degree.

Levintov Winner of the 2025 CoMSEF Postdoctoral Scholar Award for Modeling and Simulation sponsored by MSDE

Dr. Lev Levintov, Postdoctoral Research Scholar at the University of New Hampshire is the 2025 winner of the CoMSEF Postdoctoral Scholar Award for Modeling and Simulation. He is cited "***for methods development and applications of biomolecular simulations to RNA and membrane protein systems***." Prior to commencing his postdoctoral research position at UNH, Lev completed a postdoctoral position at University of Delaware (2022-23) and earned his PhD in chemical engineering at UNH (2016-21). The CoMSEF Postdoctoral Scholar Award for Modeling and Simulation recognizes outstanding research in computational molecular science and engineering, encompassing both methods and applications, among postdoctoral scholars. The award is sponsored by Molecular Systems Design & Engineering (https://www.rsc.org/journals-books-databases/about-journals/msde/), an interdisciplinary journal reporting cutting-edge molecular engineering research that is jointly owned by the Royal Society of Chemistry (RSC) and the Institution of Chemical Engineers (IChemE).

CoMSEF Business Meeting in Boston

CoMSEF will hold its annual General Meeting on Wednesday November 5 from 6-7 pm in 210 (Second Level, Hynes Convention Center). As in the past, the meeting will be held jointly with Area 1a (Thermodynamics and Transport Properties). All CoMSEF members are encouraged to attend.

Note: Opinions presented in the newsletter are the authors' and do not necessarily represent the view of the CoMSEF organization.

A Broader View Part 1: Teaching Students to Code in the Age of LLMs

Camille Bilodeau, Assistant Professor, Department of Chemical Engineering, University of Virginia

ChatGPT and other LLMs are changing the way many of us think about teaching coding-based courses. I began teaching "Numerical Methods" to undergraduate chemical engineers in January 2023, months after the first public release of ChatGPT, and quickly many of the homework problems that my course had relied upon became obsolete. Homeworks that used to involve giving students a prompt and asking them to write a code that performed a specific task quickly became trivially easy. Over the past three years, ChatGPT and other Large Language Models (LLMs) have become increasingly capable of solving coding-based tasks with less and less help from the user, leading us to wonder: what should coding-based classes look like in the age of LLMs? If LLMs are so proficient at coding, do students even need to know how to code?

I believe the answer to this question is yes and no. Decades before the advent of computers or even calculators, it was critical for students to be able to perform complex arithmetic calculations by hand. While these tedious calculations are no longer necessary, it is still important for students to learn arithmetic at a more basic level and to learn how to conceptually manipulate mathematical expressions (i.e. higher-level math). In the same way, our approach to teaching students to code must evolve from focusing on practicing basic coding skills to teaching higher order, more conceptual skills.

To determine what these higher order conceptual skills should be, we should reflect on our own experiences as chemical engineers who code as part of our research. To succeed at applying computational tools to chemical engineering problems in real-world settings, it is usually not enough to just know how to write the program. Instead, students need to know how to set up the problem and choose between multiple strategies to solve the problem. Once the code is written and implemented, it often behaves in an unexpected way, either giving an error or a suspicious output. Thus, the students also need to know how to debug their code and how to test whether their code is behaving correctly under a series of known conditions. Further, if they are operating as part of a larger team, students need to know how to organize and document their code, so that it can be usable for future applications. In this way, there are many computational skills that students need to learn beyond simply writing a program that are required for them to be proficient enough to use computational methods in a professional environment.

For those of us who code as a part of our research, we have often learned these skills through trial-and-error, rather than having been explicitly taught them. But with LLMs making coding easier, there is suddenly room in our classrooms to teach more to our students and, in turn, expect more from them. By explicitly teaching students the higher-level reasoning skills required to solve problems computationally, we can make the skills that previously required years of experience standard for all chemical engineering undergraduates.

How can we actually teach students this higher-level reasoning? Just like any curriculum re-design, this requires rethinking how the elements of our classrooms work together to attain learning objectives. This will require reflecting on our own reasoning and transforming these processes from ones that are largely guided by gut instinct to ones guided by concrete strategy. A major resource to developing these approaches will be consulting with colleagues in computer science departments, who already teach their students similar sets of skills. We then need to redesign our lectures, activities and assessments to focus on these new learning objectives.

Ultimately, LLMs aren't a threat to coding education, instead they're a prompt to raise the bar. By shifting our focus from syntax to strategy and from writing code to thinking computationally, we can prepare students to leverage these tools to become better engineers.

A Broader View Part 2: Reframing Assessment Tools in the Era of LLMs

Rose K. Cersonsky, Mike and Virginia Conway Assistant Professor, Department of Chemical and Biological Engineering, Department of Materials Science and Engineering, Data Science Institute, University of Wisconsin - Madison

Conversely, I teach Statistical Mechanics and Thermodynamics to graduate students. "Stat Mech" is a notoriously abstract subject, one that challenges students to draw from a seemingly infinite toolbox to solve complex problems. Once the right approach is identified, most problems appear trivial in retrospect - yet finding that approach is precisely where the learning occurs. Even small hints, such as which equation to use or how to frame a system, can short-circuit the process of developing conceptual understanding.

When large language models (LLMs) like ChatGPT became widely available, I worried that this delicate learning process would be lost. I imagined students struggling briefly, then turning to an LLM for the strategy, plugging numbers into the suggested equations, and moving on – never truly grasping why the approach worked or how to construct it themselves. Grading homework began to feel futile: the polished submissions no longer reflected what students actually understood. Predictably, many would later say, "I understood the homework, so I don't know why I didn't do well on the exam."

This frustration is shared among many educators, and our instinctive response has often been to restrict the use of LLMs to preserve traditional forms of assessment: homework sets, take-home exams, essays. These tools have long served as reliable indicators of student learning, yet they no longer function as intended when students have access to generative

models. The problem is not that students use LLMs, but that our assessments were never designed for a world where such tools exist.

Preparing for this past semester forced me to rethink my own approach. I came to believe that we cannot – and should not – try to prevent students from using LLMs. Instead, we must design learning experiences that remain effective even if they do. Attempts to ban these tools risk creating mistrust and, ultimately, diminish our ability to teach effectively. In statistical mechanics, the goal of a problem set is not to test algebraic manipulation, but to assess whether students can identify the essential elements of a problem and build an appropriate statistical-mechanical model. Is it a two-state system? A moving wall? How should we frame the combinatorics? To directly assess those skills, I replaced traditional homework with two new types of assignments.

First, students now design and present a homework-style problem to their classmates. They are encouraged to use LLMs for idea generation, but the accuracy and clarity of the solution are ultimately their responsibility. The results have been encouraging: in preparing to teach their peers, students spend more time thinking deeply about the assumptions and logic behind their problems. Many visit office hours to confirm their understanding, and a new kind of camaraderie has emerged in the classroom. I often hear, “Oh, you should ask X about that section – they presented a great example of it.”

Second, I now give very challenging exams but allow students to submit a revision afterward. Students can correct their tests for partial credit, explaining their reasoning and filling in gaps in understanding. This allows me to see their initial thinking under pressure, while also giving them a structured opportunity to reflect and improve. Whether or not they use an LLM in the correction process is irrelevant – the key is that they engage in conceptual reconstruction.

Encouraging deep, conceptual learning has always been a critical challenge of education This challenge is not unique to higher education or to STEM. I often talk with my mother, a sixth-grade teacher, and my sister, who teaches high school English, about how we can help students meaningfully engage in this new era. Our conclusion is always the same: rejecting new technologies has never improved learning. Instead, we must adapt creatively, using these tools to raise, not lower, the level of intellectual engagement we expect from our students.

Research Highlight: A new language for theory: Formal proofs for chemical physics

Shuwen Yue, Assistant Professor, Robert F. Smith School of Chemical and Biomolecular Engineering, Cornell University

Scientific theories are usually presented as a mix of equations and explanations, with correctness judged by human reasoning. However, even small errors in derivations or in translating equations into code can slip through review and end up in software, which results in uncertainty about whether a model is implemented to its intended logic. As scientific and engineering workflows begin to rely more on LLMs to generate code and symbolic equations, there’s a growing need for ways to automatically check that the results are logically sound. What is missing is a formal way to ensure that every step in a derivation and every line of code derived from it is correct by construction. A recent article in Digital Discovery by Bobbin, Sharlin, Feyzishendi, Dang, Wraback, and Josephson, titled *“*Formalizing Chemical Physics Using the Lean Theorem Prover*,”* addresses this need by showing how interactive theorem provers can make the reasoning behind theorems in chemical physics completely explicit and machine verifiable.[1]

The authors used the Lean theorem prover, a language for writing formal proofs, to reconstruct several fundamental theorems in chemical physics. These include the Langmuir adsorption isotherm, Boyle’s law, and the kinematic equations of motion. In each case, every assumption and algebraic transformation is written explicitly so that Lean can confirm its validity. For example, if a derivation divides by a term that might be zero, Lean stops the proof until the assumption is clarified. This process ensures that each final equation follows directly from the starting premises and that no step depends on hidden conditions. Beyond these case studies, the paper presents a broader framework for formalizing chemical theories. Within Lean, physical quantities such as rate constants, pressures, and energies are represented as structured mathematical objects that must interact in consistent ways. This approach allows proven results to be reused or extended in new derivations. The long-term goal is to build a shared library of formally verified relationships, similar to Lean’s *Mathlib* for mathematics, that provides common, error-free definitions and proofs across the physical sciences. Such a database could also serve as a benchmark for LLMs and guide AI systems to generate, translate, and verify scientific derivations within the same logical framework.[2]

Josephson’s group has since extended this framework toward molecular simulation. A recent paper in Molecular Physics introduced LeanLJ,[3] which verifies the implementation of the Lennard-Jones potential and reproduces NIST benchmark energies with full mathematical consistency. This work demonstrates that the same formal proof methods used for chemical physics equations can also be applied to the core equations used in molecular simulation. Building on this foundation, the group is developing LeanMD, a formally verified molecular dynamics framework that encodes the Hamiltonian, Newton’s equations of motion, and computable algorithms such as the Verlet integrator directly in Lean, with proofs linking each level of the simulation. These methods could lead to more reliable simulation software by embedding mathematical verification directly into the development process.

1. Maxwell P. Bobbin, Samiha Sharlin, Parivash Feyzishendi, An Hong Dang, Catherine M. Wraback, and Tyler R. Josephson. Formalizing chemical physics using the Lean theorem prover. Digital Discovery, 3(2):264–280, 2024.
2. Tyler R. Josephson. Automating Reasoning in Chemical Science and Engineering, ChemRxiv. 2025.
3. Ejike D. Ugwuanyi, Colin T. Jones, John Velkey, and Tyler R. Josephson. Benchmarking energy calculations using formal proofs. Molecular Physics, e2539421, 2025.

Where are They Now?

Now that CoMSEF has been giving the graduate student awards for more than 10 years, we've started including a "where are they now?" section in the newsletter, catching up with the winners from ~ 10+ years ago.

Yamil Colon
2014 Grad Student Award winner (Northwestern, Advisor: Randall Snurr)
Poster Title: In silico Design of Nanostructured Porous Materials for Environmental Applications

Yamil received a PhD in Chemical Engineering in 2015, under the supervision of Prof. Randall Snurr at Northwestern University. Yamil was then a postdoctoral researcher at the University of Chicago and Argonne National Laboratory until 2018, working with Prof. Juan de Pablo. There, Yamil studied self-assembly of metal-organic frameworks and worked on software for molecular simulations with enhanced sampling techniques. He joined the Department of Chemical and Biomolecular Engineering at the University of Notre Dame as the Melchor Visiting Professor in 2018 and then became an Assistant Professor in 2019. His group focuses on computational materials discovery and design using molecular modeling and machine learning tools. Applications of interest include energy storage, gas adsorption and separations, water harvesting, and photonic quantum technologies. Outside of the lab, Yamil enjoys college football, axe throwing, and traveling with his wife.

Robert Elder
2013 Grad Student Award winner (U. Colorado, Advisor: Arthi Jayaraman)
Poster Title: DNA At the Nanoscale: Interactions With Proteins, Polycations, and Surfaces

Shortly after receiving the CoMSEF graduate student award in 2013, Robert Elder received his PhD in Chemical Engineering from the University of Colorado-Boulder (2014). He went on to postdoctoral and staff positions at the Army Research Laboratory, where he studied cross-linked and semi-crystalline polymers under extreme conditions using molecular simulations. In 2019, he joined the Division of Biology, Chemistry, and Material Science in the Center for Devices & Radiological Health within the Food & Drug Administration. At the FDA, Robert has used molecular simulations and data science tools to develop methods for predicting the diffusivity of small solutes (e.g., potentially toxic leachables) in polymeric medical devices. He has published over 30 peer-reviewed articles and has helped lead the development of an FDA guidance document on the chemical characterization of medical devices. Robert lives near Baltimore, Maryland with his wife, 4-year-old daughter, 2-year-old son, and 2 cats.

We're marking this special milestone with a happy hour complete with bowling and games. Come connect, celebrate, and share in the fun with the CoMSEF community! A special thank you to the NC State College of Engineering for generously sponsoring this event. **RSVP here** to join the festivities — we can't wait to celebrate with you!

Upcoming Conferences of Interest to CoMSEF Members

AIChE Annual Meeting
Boston, MA
November 2-6, 2025
LINK

Supercomputing 25
St. Lous, MO
November 16-21, 2025
https://sc25.supercomputing.org/

ESAT 2026
Lisbon, Portugal
May 10-13, 2026
https://esat-2026.sci-meet.net/

MRS Fall Meeting
Boston, MA
November 30-December 5, 2025
LINK

ACS Spring Meeting
Atlanta, GA
March 22-26, 2026
https://www.acs.org/events/spring.html

MRS Spring Meeting
Honolulu, HI
April 26-May 1, 2026
LINK

23rd European Conference on Thermophysical Properties
Chantilly, Paris area, France
June 21-24, 2026
https://www.ectp2026.com/

FOPAM 2026
Decatur, GA
July 26-30, 2026
https://fopam.cache.org/

WATOC 2028
Merida, Mexico
January 9-14, 2028

Why CoMSEF?

Occasionally it is worthwhile to remind everyone what CoMSEF does for our community and why your membership support is important. CoMSEF was founded in 2000, and since that time it has worked to advance molecular science and engineering in diverse ways:

* We provide a forum for communication and networking within the community. The document you're reading now is a prime example, but there is more. The annual membership meeting provides a venue for communication and interaction among members. The CoMSEF web site http://comsef.org is another useful resource for this purpose. It often hosts notices about upcoming workshops, available post-doc positions, etc.

* We provide a vehicle for communication and advocacy for molecular science and engineering in relation to other research communities. For example, our four Liaison Directors identify opportunities for co-sponsorship of sessions at the AIChE Annual Meeting, facilitate programming with other organizations, and communicate and advocate CoMSEF activities with other organizations.

* We help to recognize and promote outstanding researchers and promising graduate students by funding and administering several awards. Most recently we initiated the Young Investigator Award for Modeling and Simulation. This and our other awards help the contributions of some of our best researchers to be recognized by a broad audience, extending into the larger chemical engineering community. Your dues make these awards possible.

* We provide technical programming support, ensuring we have sessions of interest to you at the AIChE meeting. These include the many sessions we sponsor or co-sponsor, as well as the CoMSEF plenary, CoMSEF poster, and Industrial Fluid Properties Simulation Challenge sessions. We also work externally to AIChE, providing technical sponsorship to conferences in our discipline (e.g., FOMMS), where we help to ensure that these events have molecular science and engineering content of the highest quality.

Your support of CoMSEF through your membership is very important in enabling us to fulfill our mission. The financial element is valuable of course, but we also gain strength in demonstrating the size of the community we represent. So please make sure to check the box to include renewal of your CoMSEF membership whenever you pay your annual dues to AIChE. When the opportunity arises, encourage your non-member colleagues in the molecular science and engineering community to join too!