Mingshuai Chen · ✉ chenms@cs.rwth-aachen.de

Lehrstuhl für Informatik 2 Softwaremodellierung und Verifikation | RWTH AACHEN UNIVERSITY

> *"Teaching to unsuspecting youngsters the effective use of formal methods is one of the joys of life because it is so extremely rewarding."*
>
> — Edsger W. Dijkstra, 1930–2002

Developing a decent series of lectures is, in my humble opinion, by no means easier than writing a research paper/proposal. Hereby with this statement, I briefly describe my overarching philosophy of teaching, my experiences in teaching and mentoring, as well as an outlook on courses that I am willing/qualified to teach in the near future.

Teaching Philosophy. My overall goal of teaching is to *assist the students in developing interests and independent, critical thinking skills for approaching fundamental principles in theoretical computer science*. In all the lectures that I am involved in, I position myself as *a coach and a co-learner* in an inclusive community consisting of me, other lecturer(s), my students, and the world around us. The topmost qualities that I value in my pedagogy include *clarity*, *preciseness*, *inspirer*, *hands-on experiences*, *individualization*, and last but foremost, *interactions*. This amounts to developing lectures with, for instance, crystal clear scopes, expectations, and grading schemas; unambiguous explanations; stimulation of creativity; pertinent (and preferably practical) exercises; tuned learning curves; and most importantly, extensive and timely interactions among the participants.

A Experiences in Teaching and Mentoring

In my early career to date, I have been involved in three different types of teaching/mentoring activities as briefed below.

Graduate-Level Courses in Theoretical CS. I have been serving as a Teaching Assistant for three graduate-level courses (as detailed below) in the context of theoretical computer science, where my primary obligation is to *deliver exercise lectures and/or laboratory sessions; develop, correct, and provide feedback on exercise and exam sheets; answer questions regularly from the class; and refer interested students to research topics/materials*.

- *Theories of Programming*, M.Sc., WS 17-18 & 18-19, UCAS. This course focuses on logic, specification, and verification, with an aim to establish a good theoretical understanding of what programs do, how they do, and why they work. Being a TA for this course was a challenging task for me: it amounted to keeping a 70-member audience —who had varying backgrounds and familiarity with the subject— interested and engrossed in a fairly steep learning curve where one may easily get overwhelmed by heavy mathematical logic and proofs. I circumvented these challenges by (1) organizing an off-the-course venue where I encouraged team working amongst participants with different backgrounds and constantly monitoring questions and difficulties raised by the students; and (2) designing laboratory sessions based on a few off-the-shelf interactive theorem provers, where the students enjoyed step-by-step, hands-on experiences and developed deeper understandings of the seemingly obscure theories behind. I was delighted to see that, by the end of the course, *over 90% of the students were able to build concrete artifacts (mechanized proofs) that they felt proud of.*
- *Theoretical Foundations of the UML*, M.Sc., SS 20, RWTH Aachen. This course aims to treat the theoretical underpinnings of major UML fragments, covering topics like formal semantics, logic, and verification. Teaching back in the year 2020 was exceptionally challenging due to the COVID-19 pandemic, and the same applies to learning of course. Together with the other lecturers, I co-developed the transformation of the course into a purely virtual format. To mitigate the effect of fewer interactions in digital teaching, I (1) exploited the tablet-pencil style of lecturing to mimic the blackboard in a physical classroom; (2) provided videocasts for all my exercise classes such that the students could repeatedly watch the videos in a preferred speed; (3) offered extra, virtual Q&A sessions to handle questions, receive feedback, and adapt the lectures (e.g., contents and pace) accordingly. I took questions from students seriously and stayed as responsive as I could. As an example, *I worked with a student towards the correction of a few critical technical flaws in the lecture contents which have been there for many years*. I was delighted to see that, in the course-evaluation report, (1) *100% of the students evaluated the lecturer as "good+", among which 37.5% rated as "very good"* (under the scale "very good - good - satisfactory - sufficient - poor"); and (2) the students commented that *"My enthusiasm to get involved with the contents of the course increased thanks to the flexible digital teaching solution." "The lecturer is always willing to help, also per email. The Q&A sessions were quite helpful."* and *"I particularly like the lecturer's way of presenting the lectures as well as his consideration of students with different background levels."*
- *Concurrency Theory*, M.Sc., WS 21-22, RWTH Aachen. This is currently an ongoing course with the aim to provide a basic understanding of modelling formalisms for concurrent systems. This is, again, a theory-oriented course and the students may quickly flinch away from the large amount of theoretical contents, e.g., logic(s), semantics, and proofs. To alleviate this problem, I constantly provide the students with hands-on experiences by designing lab sessions and practical exercises, which turns out to be quite effective according to some of the students' feedback: *"I would have*

never really understood the CCS calculus until I was able to build and run the labelled transition system underneath the example of Milner's Jobshop in [the tool] CAAL." In the meantime, I am intensively involved in *developing a hybrid teaching format that combines the strengths of interactivity in physical lecturing and flexibility in virtual teaching*. I am confident that such a hybrid solution will lead to yet another successful and decent course at RWTH Aachen.

Undergraduate/Graduate-Level Seminars in Theoretical CS. I have been supervising students through two undergraduate/graduate-level seminars (as detailed below) in the context of theoretical computer science, where my primary responsibility is to *suggest interesting topics, papers, and references; help the students to understand possibly obscure theories underneath a research paper and potential connections to related topics; provide constructive feedback to students on their report outlines, full reports, and presentations; and share with the students the best practice in scientific writings and presentations*.

- *Probabilistic Programming*, B.Sc./M.Sc., WS 20-21, RWTH Aachen. This seminar covers a variety of cutting-edge topics in the field of probabilistic programs. There, I supervised two students with topics on Probabilistic Abstract Interpretation and Slicing Probabilistic Programs, respectively.
- *Trends in Computer-Aided Verification*, B.Sc./M.Sc., SS 21, RWTH Aachen. This seminar covers the theory and practice of computer-supported formal analysis methods for both hardware and software systems. There, I supervised one student with a topic on Synthesizing Quantitative Loop Invariants for Probabilistic Programs.

Again, supervising seminar students back then was harder than usual because of the pandemic. To mitigate the negative effects, I offered to each of the students (1) a virtual kick-off meeting, (2) an online rehearsal talk, (3) immediate, itemized responses to email requests; and (4) punctuation-level proofreading. Furthermore, I assisted students to find connections between their topics and their own expertise/projects. As an example, *one student of mine developed a great interest in combining Abstract Interpretation with her Bachelor thesis on Expected Runtime of Probabilistic Pointer Programs*. During the seminars, I received a variety of positive feedback from my students reading, e.g., *"I am quite amazed by your speed [of response]; it was faster than almost all feedbacks I have received previously." "Thank you for your clarification on the definitions. This is what I suspected, but your explanation perfectly gives the reasoning for this."* and *"I have learned a lot from you about how to explore researches and how to deliver an academic talk."* Apart from formal evaluative assignments (technical reports and presentations), these feedback opportunities provided me with critical access to my students' thinking and the scope of their learning.

Mentoring Junior Graduate Students. Ever since I was a senior Ph.D. candidate at UCAS, I have been mentoring graduate students there who were junior to me. I constantly shared with them personal experiences in my early graduate years, tried to relieve them from stress and anxiety, and provided them with constructive suggestions on literature review, topic assessment, effort allocation, etc. In particular, I have been regularly assisting students to develop their immature, yet creative ideas into solid research results (which I took more like collaborations than mentor-and-mentees). As an example, *one of my mentees had co-authored two CAV papers already by the first two years of his graduate program*. Apart from the mentorship, I have been co-presenting a tutorial on Safety-Critical CPS at RTSS 2020, a top-tier conference in real-time systems, which was a splendid event to exchange insightful ideas with many graduate students in the audience from across the world.

B Teaching Plan and Interests

I am looking forward to the following teaching activity where I am scheduled in the forthcoming year.

- *Lectures within the Marie Curie MISSION Project*, WS/SS 21-22, China/Argentina. This involves secondments of researchers in the form of academic visits and/or delivery of school lectures on topics like verification, formal methods, and theories of programming. I plan to develop there a highly-interactive series of lectures on Quantitative Verification.

I find teaching extraordinarily rewarding regardless of the level or sometimes even the contents, but I have a particularly keen interest in teaching courses at the intersection of computer science and mathematics. I feel that this is where the students see and learn how theories may shape the world. I would be pleased to teach, at the undergraduate level, a wide range of courses in areas of, e.g., Automata Theory, Compiler Construction, Programming Languages, Software Engineering, Probability Theory, and Discrete Mathematics. At the graduate level, I am willing to teach courses like Mathematical Logic, Theory of Computation/Programming, Program Analysis and Verification, Computer Algebra, and Concurrency Theory. Moreover, I have enough experience in Algorithms, Computer Architecture, and Databases that I can productively co-teach these courses with faculty colleagues.

In addition, I am particularly interested to develop the following new advanced graduate-level courses and seminars.

- *Modelling and Verification of Probabilistic Systems*. This course aims to address compositional modelling and automated verification (i.e., model checking) of probabilistic models, covering topics like Markov chains, Markov decision processes, probabilistic automata, Markov automata, model checking, Bayesian inference, probabilistic temporal logic, bisimulation, compositional modelling, and concurrency. I intend to build the course upon prior experience gained at RWTH Aachen, and further develop it by incorporating the recent burst of probabilistic verification techniques, e.g., k-induction, invariant synthesis, probabilistic abstract interpretation, etc.
- *Foundations of Cyber-Physical Systems*. I intend to deliberately develop this course to expose the students to the core semantics and fundamental principles towards the design, verification, synthesis, and abstraction of cyber-physical systems. I foresee that materials from similar courses by R. Majumdar at MPI-SWS and A. Platzer at CMU could be of great help, on top of which I plan to introduce fresh topics like controller synthesis and system identification.
- *Seminars on Recent Advances in Verification and Synthesis*. I would like to regularly offer seminar positions for students who are interested in the latest advances in computer-aided verification and/or synthesis. I expect this series of seminars not only to draw the students' research interests in these fields, but also to deliver the best practice in scientific readings, writings, and presentations.

I am, of course, open and keen to discuss with faculty colleagues for a more refined and tailored teaching program in my early appointed years.

C Conclusion

It has been my long-held belief that universities and research institutes have an important social responsibility of training and mentoring students to have open minds and to think rationally. My personal experiences in teaching and mentoring in the past few years laid my confidence in contributing to the loop. The process of teaching and mentoring, in turn, rewards me with the pleasure to understand, learn from, and mould young minds towards excellence. Therefore, *I take the liberty to consider myself as a broadly trained computer scientist who is capable of complementing the existing teaching and research strengths of the College of Computer Science and Technology at Zhejiang University.*