Teaching statement
Raja R. Sambasivan

As a systems and networking researcher, I deeply enjoy helping students unravel the complexity of the computer systems we depend on in our everyday lives. As a result, I am interested in teaching (and qualified to teach) the following classes at the undergraduate and graduate levels: cloud computing, distributed systems, operating systems, networking, and storage systems, as well as special topics courses on my research. To promote interest among a wide range of diverse students and to discourage hacker culture, my teaching philosophy involves first emphasizing the benefits to society of computer systems’ constructs (e.g., how low-latency networking techniques enable remote surgery to connect expert surgeons with critically-ill patients who can’t be moved). I then discuss how the constructs work, the design decisions involved in creating them, and other possible approaches. Often the right answer to how to design a system is “it depends.” My goal in teaching is to empower students with the ability to identify and systematize design choices, to identify tradeoffs between them, and to pick design choices that will yield simple, efficient, and robust systems for different environments and applications.

Teaching classes: In the Fall of 2013, as a postdoctoral researcher, I developed and co-taught the inaugural version of Carnegie Mellon’s (CMU’s) graduate class on cloud computing (15-719). The syllabus we developed, which was suited for Masters’ students and first or second year PhD students, provided students with a broad overview of all the topics that fall under the cloud-computing umbrella (e.g., virtualization, distributed storage, and techniques to to diagnose problems in clouds). I created and delivered approximately one-third of the lectures and created and supported both of the class’ projects. The projects aimed to give students experience with using clouds and building them. Therefore, the first project required students to write map-reduce jobs within AWS and the second required them to build a load balancer in OpenStack. Due to the success of the inaugural version and students’ strong interest in the material, the class has been re-offered every year since 2013. I was invited to deliver guest lectures on problem diagnosis for these subsequent iterations for the duration of my stay at CMU.

During the Spring of 2017 and the Spring of 2018, as a research scientist at Boston University (BU), I delivered guest lectures on networking and diagnosis in BU’s graduate cloud-computing class (EC 328). For the Spring 2018 iteration, I mentored two groups of students on semester-long projects, which are an integral part of this class. These projects focused on integrating an open-source tracing infrastructure (Jaeger) into various open-source distributed applications (e.g., Ceph). As a graduate student at CMU, I served as a TA for CMU’s graduate storage systems class twice.

I teach using the Socratic method, which engages students in a dialogue that guides them toward self-discovery of concepts and ideas. For example, I constructed my cloud computing lecture on problem diagnosis around an interactive discussion of an AWS outage that had been written about extensively. During the lecture, I presented each step AWS engineers took to diagnose the problem and asked students to brainstorm whether a given step improved the situation or worsened it. This discussion led to how the research papers students had read as part of their assigned readings could have helped (or not helped) engineers avoid missteps. I framed the lecture as a “whodunit-style” mystery, which I found greatly increased student engagement.

Advising: I have been extremely fortunate to work with many students. As part of helping with the initial stages of a new research lab at BU (the Mass Open Cloud), I am co-advising (or previously co-advised) six students on projects related to building diagnosis tools for cloud ecosystems. I am also helping advise (or previously helped advise) two students on datacenter networking and big-data analysis frameworks. Many of these students are junior graduate students and four are women. I previously mentored one student during my postdoctoral research and several during my graduate studies.

While my approach for advising varies per student, there are certain overarching themes. Most importantly, advising involves being part cheerleader and part mentor. With regard to the former, I try to help students maintain their drive by instilling a sense of ownership, always being positive, and helping them weather the lows that are a natural part of being a researcher. With regard to the latter, I encourage students to identify solutions to problems independently, but provide guidance (as necessary) along the way. For junior students especially, I encourage both frequent informal communication (e.g., over slack) and structured weekly meetings during which I provide detailed feedback. It is often challenging for junior students to understand how much progress they should make before asking for feedback; these informal communications and meetings help establish these expectations. During meetings, I focus on fully understand what students are trying to communicate. In cases where I disagree with students’ approaches, I ask probing questions rather than assert my own worldview. I have found that such probing often leads to new insights both for students and for the advisor.

Outreach: I am deeply interested in increasing school-age children’s interest in STEM fields. My current efforts include advising high-school students on research projects as part of the MIT primes research program and serving as a pen pal to underprivileged middle-school children. The two high-school students I mentored in 2017 and 2018 were named Siemens Competition Semi-finalists in 2017 and presented their work at a Red Hat developer conference in 2018. In the past, I volunteered in a science center to communicate my love of astronomy to grade-school children.

December 2018