



CS Talk Series

The Automated-Reasoning Revolution: From Theory to Practice and Back

Moshe Y. Vardi (Rice University)

Host: Tom Henzinger

For the past 40 years computer scientists generally believed that NP-complete problems are intractable. In particular, Boolean satisfiability (SAT), as a paradigmatic automated-reasoning problem, has been considered to be intractable. Over the past 20 years, however, there has been a quiet, but dramatic, revolution, and very large SAT instances are now being solved routinely as part of software and hardware design. In this talk I will review this amazing development and show how automated reasoning is now an industrial reality. I will then describe how we can leverage SAT solving to accomplish other automated-reasoning tasks. Counting the number of satisfying truth assignments of a given Boolean formula or sampling such assignments uniformly at random are fundamental computational problems in computer science with applications in software testing, software synthesis, machine learning, personalized learning, and more. While the theory of these problems has been thoroughly investigated since the 1980s, approximation algorithms developed by theoreticians do not scale up to industrial-sized instances. Algorithms used by the industry offer better scalability, but give up certain correctness guarantees to achieve scalability. We describe a novel approach, based on universal hashing and Satisfiability Modulo Theory, that scales to formulas with hundreds of thousands of variables without giving up correctness guarantees.

Wednesday, October 18, 2017 01:00pm - 02:30pm

IST Austria Campus Mondri Seminar Room 2, Central Building



This invitation is valid as a ticket for the IST Shuttle from and to Heiligenstadt Station. Please find a schedule of the IST Shuttle on our webpage (note that the IST Shuttle times are highlighted in dark green): http://ist.ac.at/fileadmin/user_upload/pdfs/IST_shuttle_bus.pdf
The IST Shuttle bus is marked IST Shuttle (#142) and has the Institute Logo printed on the side.