CSCI 599: Formal Verification of Computer Systems

Units: 4

Spring 2017 – Mon, Wed—10:00pm – 11:50pm

Location: GFS 207

Instructor: Chao Wang

Office: SAL 334
Office Hours: TBD
Contact Info:

Http: http://www-bcf.usc.edu/~wang626/
Email: wang626@usc.edu

Teaching Assistant: TBD

Office: TBD
Office Hours: TBD
Contact Info: TBD

Course Description

This course covers techniques for formal specification and verification of computer systems. Topics include temporal logic, model checking, abstract interpretation, and symbolic decision procedures such as binary decision diagrams (BDDs), Boolean satisfiability (SAT) solvers, and satisfiability modulo theory (SMT) solvers.

Learning Objectives

Students who take this course will gain an understanding of the concepts and theories of computer-aided formal verification, and learn how to use and write formal verification tools. Major learning objectives are:

1. Write formal properties and specifications in computational tree logic (CTL),
2. Write formal properties and specifications in linear-time temporal logic (LTL),
3. Verify systems using CTL model checking and LTL model checking,
4. Scale up verification using automated abstraction and refinement,
5. Construct and use BDDs in symbolic model checking,
6. Write and use SAT/SMT solvers in bounded model checking,
7. Use abstract interpretation to generate program invariants.

Prerequisites

1. General proficiency in discrete mathematics
2. C/C++ programming skills

Course Notes

N/A

Required Readings and Supplementary Materials

There are no textbooks. Lecture notes will be provided by the instructor. In addition, the following books are supplementary reading materials (optional):

1. Model Checking, by Clarke, Grumberg, and Peled, The MIT Press, 1999.
2. The Calculus of Computation: Decision Procedures with Applications to Verification, by Bradley and Manna, Springer 2007.

Description and Assessment of Assignments

The grades will be based on the completion of six homework assignments, a midterm exam, and a final project. The breakdown for each of these categories is listed below. A more detailed explanation of each category is also provided.

Homework

Homework includes a programming related assignment (HW0, which accounts for 10% of the grade) and five standard assignments (HW1-5, each of which accounts for 5% of the grade). Homework assignments must be submitted electronically. Please take care to upload the correct files, because they are the ones that will be graded.

Midterm exam

Midterm exam will be open-book and open-notes, but electronic devices will not be allowed. There will be no make-up exams. If you have an important reason for missing the midterm, please make arrangements with the instructor in advance. Otherwise, a missed exam receives a grade of 0.

Research paper presentation

Research paper presentation asks each student to choose a paper from a list of recently published papers suggested by the instructor. The goal is to teach others about the topic, so in the end, everyone in the class will have a better understanding of the most recent development in the field. To get a good grade in paper presentation, you need to excel in the following aspects:

- Clarity in presentation (how well you understand the paper and handle questions, how smooth your talk is, etc.)
- Quality of the slides (your slides must to be informative and thorough, with technical depths, figures, etc.)

Final project

Final project is a research project that asks students to develop new formal verification techniques or identify innovative uses of existing formal verification techniques. At the end of the project, each student must submit a project report. Your grade on the final project depends on the following aspects:

- Novelty of the project design,
- Thoroughness in the execution, and
- Clarity in the project report.
-

Below are two sample final projects:

1. *Project Summary – Formal Verification***:** In this project, students are required to find an interesting application, formulate it into a formal verification problem, and solve it using techniques learned from this course. For example, you may use any of the following verification tools, such as NuSMV (http://nusmv.fbk.eu/NuSMV/), CBMC (http://www.cprover.org/cbmc/) or VIS (http://vlsi.colorado.edu/~vis/).\
2. *Project Summary – Program Synthesis:* Syntax Guided Synthesis (SyGus) is an emerging technique for directly generating software code from a set of logical specifications. In this project, students are required to learn the SyGuS specification language and the basics of SyGuS tools. You need to identify an interesting application of your choice, formulate it into a SyGuS problem, and solve it using a SyGuS tool (e.g., http://www.sygus.org/).

Grading Breakdown

Assignment	**% of Grade**
Homework	35%
HW0 programming related (10% of the grade)	
HW1-5 standard (5% of the grade each)	
Midterm exam	25%
Research paper presentation	10%
Final project	30%
TOTAL	**100%**

Additional Policies

Late assignments will be accepted up to 24 hours after the announced deadline, with a penalty of 20%. Assignments received more than 24 hours late will receive a grade of 0.

If you feel that an error has been made in grading, please notify the grader within one week after the material is returned. For exams and final projects, please present a short written appeal to the instructor.

Course Schedule: A Weekly Breakdown

	Topic	Reading	Slides	Assignment
Week 1	Overview/Admin Kripke structure, CTL	Chapter 1 Ch.2_A	Lecture_0 Lecture_1	HW0 out
Week 2	CTL model checking	Ch.2_B	Lecture_2 Lecture_3	HW0 due
Week 3	Fairness constraints Counterexamples	Ch.2_C	Lecture_4 Lecture_5	HW1 out
Week 4	Simulation relations	Ch.3	Lecture_6 Lecture_7	HW1 due
Week 5	Abstraction refinement	Ch.3	Lecture_8 Lecture_9	HW2 out
Week 6	LTL model checking	Ch.4	Lecture_10 Lecture_11	HW2 due
Week 7	LTL and omega automata	Ch.4	Lecture_12 Lecture_13	HW3 out
Week 8	Binary Decision Diagrams **(Midterm)**	Ch.5	Lecture_14	(Project proposal due) HW3 due
Week 9	Symbolic model checking	Ch.6	Lecture_15 Lecture_16	HW4 out
Week 10	SAT solvers	Ch.7	Lecture_17 Lecture_18	HW4 due
Week 11	Bounded model checking	Ch.8	Lecture_19 Lecture_20	HW5 out
Week 12	SMT solvers		Lecture_21 Lecture_22	
Week 13	Advanced topics: Abstract interpretation	Recent papers	Student presentations	HW5 due
Week 14	Advanced topics: Controller synthesis	Recent papers	Student presentations	
Week 15	Advanced topics: Program synthesis	Recent papers	Student presentations	
FINAL	**(Project report due)** no exam			**(Project report due)**

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Part B, Section 11, "Behavior Violating University Standards" https://policy.usc.edu/student/scampus/part-b. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Discrimination, sexual assault, intimate partner violence, stalking, and harassment are prohibited by the university. You are encouraged to report all incidents to the *Office of Equity and Diversity/Title IX Office* http://equity.usc.edu and/or to the *Department of Public Safety* http://dps.usc.edu. This is important for the health and safety of the whole USC community. Faculty and staff must report any information regarding an incident to the Title IX Coordinator who will provide outreach and information to the affected party. The sexual assault resource center webpage http://sarc.usc.edu fully describes reporting options. Relationship and Sexual Violence Services https://engemannshc.usc.edu/rsvp provides 24/7 confidential support.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://ali.usc.edu, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://dsp.usc.edu provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* http://emergency.usc.edu will provide safety and other updates, including ways in which instruction will be continued by means of Blackboard, teleconferencing, and other technology.