Trustworthy Autonomy

General Course Information

- *Term:* Spring 2024
- *Department*: Computer Science
- *Course Number:* 691-002
- *Time:* Mon & Wed, 3:30 pm - 4:45 pm
- *Location*: Hardaway Hall 251
- *Course Webpage:* bineet.cs.ua.edu/teaching/S2024/CS691/TrustworthyAutonomy.html (Instructor will make his best effort to post the key dates, lecture slides, *etc.*)

Instructor

Dr. Bineet Ghosh

- *Email:* bineet@ua.edu
  - Please add the following in subject line if you are emailing regarding the course: `[S24-CS691] <your-subject>`
- *Office:* CYB 3053
  - *Office Hours:* Mon & Wed, 1:30 pm - 2:30 pm, or by appointment (bineet@ua.edu).
- *Phone:* 205-348-6064 (This is office phone, and may not always be accessible)
- *Instructor Webpage:* bineet.cs.ua.edu

Course Description

Last decade has a significant growth in deployment of autonomous systems in several places, such as, driving, factory floors, surgeries, wearables, and home assistants, *etc.* As a result of such diverse areas of its deployment, autonomous systems are required to operate in a wide range of environments with uncertainties (*viz.*, sensor errors, timing errors, dynamic nature of the environment, etc). Such environmental uncertainties, even when present in small amounts, can have drastic impact on the safety of the system. In this course, we shall discuss formal techniques that are able to verify and design safe autonomous systems, even in the presence of such uncertainties, allowing for their trustworthy deployment in the real world.

Design and Verification of Autonomous Systems lies at the intersection of Formal Methods, Real-Time and Embedded Systems, and Control Theory, with practical applications in diverse domains such as Robotics, Automotive, Industrial and Home Automation Systems. *Importantly, prior knowledge of Formal Methods, Real-Time and Embedded Systems, or Control Theory is **NOT** a prerequisite for this course, as we will cover the fundamental concepts during the class.* The main objective of this course is to equip the students with the knowledge of techniques that guarantee performance and safety of autonomous systems, instilling trust in their deployment, especially in critical safety-critical situations.

Prerequisites

1. Algorithms (CS 201/470)
2. Theory of Computation (CS 475)
3. Mathematical Maturity
4. Coding in Python (Or, strong background in other languages and willing to learn Python)

Speak to instructor, if unsure.

Textbooks

NO textbooks are required, but following textbooks can be very beneficial.

- Formal Methods:
  1. Principles of Model Checking
  2. Decision Procedures
- Control Systems
  3. Linear Systems Theory
  4. Principles of Cyber-Physical Systems
  5. Introduction to Embedded Systems

Course Objectives

1. Introduction to Formal Methods:
   - Understand the fundamental concepts and principles of formal methods.
   - Gain insights into the techniques used for verifying the safety and correctness of programs and autonomous systems.
   - Explore the theoretical foundations of formal methods and their practical applications.
2. Autonomous Systems:
   - Understand the concept of autonomous systems.
   - Learn to model autonomous systems for formal analysis.
   - Learn about different types of models and their inherent properties.
3. Design and Verification of Autonomous Systems:
   - Apply the formal methods introduced in the first part of the course to design and verify autonomous systems.
   - Explore real-world applications of safe autonomous systems, with a focus on automotive and robotics.
   - Review research papers related to formal methods in autonomous systems.
   - Engage in a course project that involves the design, verification, and practical implementation of safe autonomous systems, using knowledge from the course.

Student Learning Outcomes

- Being able to understand and use different types of formal modeling of autonomous systems.
- Being able to understand and formalize safety and performance for given system in a given application scenario.
- Being able to understand and apply formal techniques to prove correctness, and design safe systems.

Outline of Topics

- Introduction
    1. Autonomous systems
    2. Notion of trust in deployment of autonomous systems
- Formal Methods
    3. Mathematical logic
    4. Verifying correctness of programs
    5. Software testing
- Cyber-Physical and Autonomous Systems
    6. Introduction to control theory
    7. Linear systems
    8. Hybrid automata
- Verification of Autonomous Systems
    7. Understanding uncertainties
    8. Provable safety verification of autonomous systems
    9. Designing provably safe and performance optimal systems

Grading

- Homework (minimum 4): 30%
- Paper presentation, and leading class discussion (1-3 papers): 30%
- Final project: 30%
- Class participation & student's growth: 10%

Student's growth. Identifies the effort being made by a student to understand the concepts being taught, despite their prior backgrounds. In other words, this can be used identify the efforts being put in by a student even if it is not being reflected in their scores. Discussing with instructor (especially during office hours) is a good way to demonstrate that.

Late work policy

- One late homework submission will have 0 (zero) penalty, if submitted within a week of the posted deadline.
- Any further late submission will result in the following deduction scheme: $min(15 \times (\text{no-of-days-late}), 100)\%$ will be deducted.

- Final project submission date is strict. Failure to submit the report on time will result in 0 score.
- In case of emergency (or unforeseen cases), please speak to instructor. The instructor will try his best to accommodate for the situation (subject to proof) in accordance with the university policy.

Final project

- **Types**: Survey (literature review), Application, Theory.
- **Logistics**
    - Maximum of three people in a group.
        - Each group member is responsible to *clearly* identify their contribution.
    - 1-3 reports maybe due before the final report and presentation to share progress with the class.
    - *Presentation:* 20-30 mins, followed by discussion, debate, comparative analysis.
    - *Report:* 8 page document in IEEE conference format (similar to conference paper).

Typical Class Activity

The class activities are designed to promote active learning, critical thinking, and a understanding of the challenges in the design and verification of autonomous systems.

- **Lectures on Fundamental Topics.** The initial portion of the course focuses on building a foundation by covering fundamental concepts and principles related to the design and verification of autonomous systems. These lectures provide students with essential theoretical knowledge and practical insights, ensuring they have the necessary background to engage effectively with advanced topics and projects later in the course.
- **Student Paper Presentations.** Each student selects and presents one or more research papers relevant to the course. Following the presentation, there will be discussion, debate, comparative analysis (Note: this is the best way to demonstrate class participation and student's growth).

Attendance and Participation

- Though attendance will not be formally recorded, it is recommended the lectures are attended. Please note that lectures are the best way to demonstrate the student growth (which has 10%).
- Further, if a student has missed noticeably many lectures (for instance, shows up only during their presentation), there will be a minimum of 50% deduction from their obtained score in the student’s growth section.
- Strongly recommend visiting during the office hours at least three times during the semester.

Disclaimer Notification of Changes

The instructor will make every effort to follow the guidelines of this syllabus as listed; however, the instructor reserves the right to amend this document as the need arises (including project due dates and test dates). In such instances, the instructor will notify students in class and/or via email and will endeavor to provide reasonable time for students to adjust to any changes.

Standard Course Policies

Please refer to the university's standard course policies (Honor code, misconduct, behavior etc.) for details.