Complex Adaptive Systems

UNM CS 423/523 Spring 2019
Tu, Th 9:30 – 10:55
Collaborative Teaching and Learning Building room 210
Website: https://moseslab.cs.unm.edu/courses/sp2019-cs523/index.html
Piazza site: https://piazza.com/unm/spring2019/cs523/home

Professor: Melanie Moses
Email: melaniem@cs.unm.edu
Office Hours:
Tu 10:45-11:45pm after class in CTLB
Wed 11:00-12:00am, Farris 3340

TA: Akil Andrews
Email: akilandrews@unm.edu
Office hours by appt Farris 3320

Textbook: We don't have a textbook for this course, but we'll use as a guide *Complexity, A Guided Tour*, Melanie Mitchell, Oxford University Press. Each week we'll read journal articles to explore concepts from the book in more depth. These articles will include more detailed technical explanations and recent conceptual advances in understanding CAS.

Course Description: A graduate level introduction to selected topics in complex adaptive systems (CAS). The course focuses on computational tools to simulate and measure complexity, and analysis of biological and computational complex adaptive systems. Topics include definitions of complexity, evolution and genetic algorithms, cellular automata, dynamical systems, scaling and fractals, ant colonies & ant colony optimization algorithms, immune systems & immune inspired computer security and swarm robotics.

Course Assignments and Grading:

The course requires extensive reading, participation in discussions and in-class exercises, attending lectures, and completing programming assignments, written reports and oral presentations. In addition to the primary textbook, students will read papers from the primary scientific literature or chapters from supplemental textbooks each week.

• 20% of the course grade will be based on class participation, including short pop quizzes to ensure that you have completed the reading, participation in in-class exercises, and leading and participating in discussions of assigned readings. Students enrolled in the graduate section (CS 523) will work in teams to present a paper to the class and lead a discussion on that paper. The oral presentation (usually with slides) will include a paper summary and questions to facilitate discussion. Students must meet with the professor beforehand to review their presentation.
• 20% of the grade will be based on a midterm.
• 60% of the grade will come from three reports based on programming projects. The first project will be relatively simpler (a simple set of programs, data analysis and presentation

in a written report). The second and third projects will be more complex. Students may come up with their own independent project for the third project, but they must have the project topic approved by the professor beforehand.

You will work in pairs for all assignments and document the contribution of each team member. For each project, you will turn in your code, a readme file describing how to run your code, and a report describing how your code works, results and analysis and answers to specific questions. Your grades will be based primarily on the quality of your reports which should be clear, concise, free of typos and grammatical errors and contain clear and meaningful figures. Your reports should indicate an understanding of relevant concepts covered in lectures, readings and discussions. You should spend at least as much time writing your report as writing your code.

Projects turned in late will be penalized 10% for each late day, for a maximum of 3 days. Students who have a true emergency must contact the professor before the due date. No exceptions.

Academic dishonesty will not be tolerated. If you cheat, you will fail the class. In collaborative work, the contributions of each student must be documented clearly in an author contributions section of the report. Your report must clearly document all downloaded code and how you have modified or incorporated it into your own code. Failure to document the source of any code that you did not write yourself constitutes cheating. Similarly, you must cite all journal articles, books, web pages and other online sources for your reports in a references section.

Course Topics

Introduction & Foundations: Definitions of complexity; dynamical systems, information theory
Evolution and Genetic Algorithms
Cellular Automata
Swarm Robotics
Natural and Computational Immunology
Midterm Review & Midterm
Brains, Neural Nets & Analogies
Modeling & the Prisoner's Dilemma
Networks, scaling & fractals
Complexity Revisited

***Tentative, Approximate* Due Dates**

Project 1 part 1 due: Jan 31th
Project 1 part 2 due: Feb 19th

Midterm, March 21st

Project 2 due: March 28

Project 3 Independent Project Proposals: April 4
Project 3 Final Paper: May 2nd (11:59 pm)
Project 3 **Presentations**: in-class April 25 - May 2