BIO-INSPIRED COMPUTING
ASU CSE 598
Spring Term, 2020
Mon, Wed 3:05 - 4:20 pm BYAC 270

Professor: Stephanie Forrest

Office: BYENG 394, Biodesign Bldg. B 120A
Phone: 480-727-0492 (during office hours only)
Email: steph@asu.edu (anytime)
Office Hours: Mon. 2-3 (BYENG), Wed. 11:00-12:00 (Biodesign), or by appointment

Textbook: Selected Readings and optional textbooks:
M. Mitchell *An Introduction to Genetic Algorithms* MIT Press 1998.
L. Sompayrac *How the Immune System Works* (Blackwell Publishing, multiple editions, I like the 2nd edition the best)
D. Gordon *Ant Encounters* (Princeton, 2010)

Course Description

A graduate introduction to bio-inspired computing, discussing computational methods that are derived from biological processes and models. No prior knowledge of biology is expected. Basic knowledge of computer science, programming competence, and intellectual maturity are expected.

The course will cover computational algorithms, models, and applications that are inspired from three primary areas of biology: evolution, immunology, and social insects. It will discuss some other recent applications of biology in computing, including metabolic scaling theory, molecular computation, epidemiology, and collective computation. The course will not cover neural systems to avoid overlaps with existing courses in machine learning and neural networks. Each topic will introduce the relevant biological concepts and important applications of the concepts in computing, discuss relevant mathematical models and analyses, and read one or two current research papers that are relevant to the topic.

There will be some assigned programming projects, regular reading assignments and discussion, and a student-initiated final project. Students are expected to write and run computer programs to complete the assignments. Biological concepts will be presented in lecture accompanied by introductory readings. In addition to lectures, we will read one to two papers per week, and there may be some small writing assignments and oral presentations.

I. Course Introduction (1 week) Why is biology relevant to computer science and why is computer science relevant to biology? The role of abstraction and modeling. Evaluation metrics. Overview of topics and expectations.

II. Evolution (4 1/2 weeks)

1. Biological underpinnings (1 lecture)
2. Applications in computing and engineering (3 lectures / discussion)
   (a) Evolutionary computation: Genetic algorithms, genetic programming, evolutionary computation as scientific models
   (b) Mathematical Models: Search spaces and biased sampling, multiplicative weights update algorithm, what is learnable by selection and mutation?
3. Current research (3 lectures / discussion)
   (a) Repairing and improving software
   (b) Mutational robustness, novelty search
   (c) Automatically evolving neural network architectures
   (d) Evolving strategies to evade censorship

III. Immunology (2 1/2 weeks)

1. Biological underpinnings (1 lecture)
2. Applications in computing and engineering (2 lectures /discussion)
   (a) Negative selection algorithm
   (b) Architecture for an artificial immune system
3. Current research (2 discussions / student presentation)
   (a) Collective decision-making of T-cells
   (b) Distributed nestmate recognition
   (c) Other applications to computer security

IV. Social Insects (2 weeks)

1. Biological underpinnings (1 lecture)
2. Applications to computing and engineering (2 lectures / discussions)
   (a) Ant Colony Optimization (ACO) algorithms
   (b) Robotics and other multi-agent systems
3. Computational models and theory (1-2 discussions / student presentation)
4. Current research (1-2 discussions / student presentation)

IV. Other topics (3 weeks)

1. Metabolic scaling theory (1 lecture / discussion, student presentation)
   (a) Predicting power consumption on chips
2. Molecular programming (1 discussion, student presentation)
   (a) Building neural network algorithms in DNA
3. Epidemiology and Ecology (1 discussion / student presentation)
   (a) Network topology and susceptibility to attack
   (b) Red Queen dynamics and arms races in cybersecurity
   (c) Epidemic spreading
4. Collective computation (2 discussions / student presentation)
   (a) Theories of collective computation
   (b) Examples of collective computation

V. Student Presentations and course wrapup (1 1/2 weeks)