Nature-Inspired Search and Optimisation
Advanced Aspects of Nature-Inspired Search and Optimisation (Ext)

Shan He

School for Computational Science
University of Birmingham

Module 06-28209, 28211, 27818 and 27819: Nature-Inspired Search and Optimisation
Advanced Aspects of Nature-Inspired Search and Optimisation (Ext)

Outline of Topics

About this module

Why study Nature Inspired Optimisation

The modules

Level 3/H

20 credits

- 2-hr lab session per week (Shan He)

50%:

10 credits

- two 1-hr lectures per week

Level 4/M

20 credits

- 2-hr lab session per week (Shan He)

50%:

10 credits

- two 1-hr lectures per week

Lectures times/venues

- Lectures time and location
  - Tuesday 10:00am Arts Building Lecture Room 1 (125)
  - Thursday 13:00pm in Sports and Exercise Sciences Lecture Theatre
  - Thursday 15:00pm in Computer Science Building LG04 Lab
- Please feel free to ask me questions:
  - Email: s.he@cs.bham.ac.uk
  - Office hour: Thursday 17:00pm

Exam and Continuous Assessment

Level 3/H

20 credits

- lab-based assignments (50%) – information provided by Shan He

50%:

10 credits

- 1.5 hr exam (90%)
- fortnightly homework assignments (10%)

Level 4/M

20 credits

- lab-based assignments (50%) – information provided by Shan He

50%:

10 credits

- 1.5 hr exam (80%)
- fortnightly homework assignments (10%)
- 'research' essay (10%)

Aims of the module

We will learn:

- For which types of problems should we use nature-inspired search and optimisation methods?
- What are common design elements?
- What are basic principles for their design?
- How can we categorise different types of these methods?
- How can we select one that fits our problem well?

Learning outcomes

- All versions
  - Describe different nature-inspired search and optimisation methods and explain how they are applied to solve real world problems.
  - Discuss relations, similarities and differences between the most important heuristics and nature-inspired algorithms presented in the module and other search and optimisation methods.
  - Design and adapt nature-inspired algorithms including operators, representations, fitness functions and potential hybridisations for non-trivial problems.
- Advanced version (20 credits)
  - Implement nature-inspired algorithms using different programming languages and compare them experimentally.
- Level 4/M
  - Describe, use, analyse and discuss recent research literature in a sub-field of natural computation and demonstrate a critical understanding of these methods.

What will be covered (**Could be changed later**)

- Algorithms
  - Randomised algorithm and Randomized Search Heuristics
  - Evolutionary Algorithms: Genetic Algorithms, Genetic Programming, Evolutionary Programming, Differential Evolution and Evolution Strategies
  - Swarm Algorithms: Particle Swarm Optimiser and Ant Colony Optimisation
  - Artificial Immune System
- Theory
  - Schema Theorem
  - Convergence and Convergence Rate
  - Computational Complexity
  - No Free Lunch Theorem
  - Fitness Landscape

How the 2 hours lab are organised

- We will focus on real-world problems, e.g., travelling salesman and crew scheduling problems.
- I will present and explain some **code examples**
- You can try those code examples and complete **some exercises** to solve the problems.

Questions and answers

- Q: What is Nature Inspired Optimisation?
- A: The study of computational systems that use ideas inspired from natural evolution, e.g., the principle of survival of the fittest
- Q: Why study it?
- A: It provides a **general** method for solving 'search for solutions' type of problems, such as optimisation, learning, and design.

Cool example 1: NASA's Evolved antenna

- Satellite antenna evolved fully designed by an evolutionary algorithm
- Used for a 2006 NASA mission called Space Technology 5 (ST5)
  - The mission consists of three satellites that will take measurements in Earth's magnetosphere.
  - Each satellite has two communication antennas to talk to ground stations.
  - The mission successfully launched on March 22, 2006
- It is the the world's first artificially-evolved object to fly in space
- Youtube video

Cool example 2: Evolutionary robotics

- Evolutionary robotics: a methodology that uses evolutionary computation to design, develop and control autonomous robots.
- One of the famous example: The Golem Project
  - Genetically Organized Lifelike Electro Mechanics
  - H. Lipson and J. B. Pollack (2000), "Automatic design and Manufacture of Robotic Lifeforms", Nature 406, pp. 974-978
  - Evolutionary Computation + 3D printing = Evolutionary Robots
- Youtube video
- Youtube video of new evolving soft robots

Cool example 3: Blondie24 - Evolved master checkers player

- Blondie24: an artificial intelligence checkers-playing computer program
- The most special feature: learn playing checkers by itself
- How: given the checkers rules, co-evolve artificial neural networks as opponent players, the wining players will be selected and breed the next generation
- Did it work? Rated as Expert on Microsoft's Gaming Zone as claimed by the authors.
- Video

Cool example 4: Automated Invention Machine

- An ambitious goal: To automate creativity
- How: use evolutionary computation, especially Genetic Programming
- A case study: invent/re-invent electronic circuits by Genetic Programming
- Reinvented 21 Previously Patented Inventions
- Invented 2 new patentable inventions
- John Koza's website

Cool example 5: Evolutionary art

- William Latham - Mutator 1 + 2 : Evolutionary Art
- Music evolved by Evolutionary Computation