Course Description

This course provides the foundations to Evolutionary Computation, emphasizing in the main meta-heuristics used in the field: Genetic Algorithms, Evolutionary Strategies, Particle Swarm Optimization, Differential Evolution, and Compact Genetic Algorithms. These Artificial Intelligence paradigms focus on solving optimization problems using stochastic population-based meta-heuristics. These techniques have been used to solve many engineering problems; some of them will be studied in class. A first programming project will involve developing a Genetic Algorithm (on the language of your choice). On subsequent projects, students may choose to continue their own implementations or use public domain software packages (e.g. ECJ, Evolvia, CILib, ECsPy, etc.)

Schedule

<table>
<thead>
<tr>
<th>Topic</th>
<th>Date</th>
<th>Reading</th>
<th>Grade</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>9/11</td>
<td>1.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Genetic Algorithms</td>
<td>9/18</td>
<td>1.1, 1.2</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Review</td>
<td>9/25</td>
<td>1.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Evolution Strategies</td>
<td>10/2</td>
<td>2.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Particle Swarm Optimization</td>
<td>10/9</td>
<td>3.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Compact Genetic Algorithms</td>
<td>10/16</td>
<td>4.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Differential Evolution</td>
<td>10/23</td>
<td>5.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Presentations</td>
<td>10/30</td>
<td>6.1</td>
<td>95%</td>
<td>Code example</td>
</tr>
<tr>
<td>Final Exam</td>
<td>11/7</td>
<td></td>
<td></td>
<td>Final Exam</td>
</tr>
</tbody>
</table>

Grading

- **Recommended Prerequisites**: Artificial Intelligence, Programming Languages.
- **Grading**:
  - Programming projects 80%
  - Presentations/participation 20%

References

Introduction
"For thousands of years, people have used crossover and selection to get better crops, horse breeds, or roses. Nevertheless, translating these procedures to used them in computer programs is not easy. The main problem is the construction of a genetic code capable of representing the structure of different programs in the same way DNA represents the structure of a person or a mouse."  
John Holland, 1992

Evolution Theory
- (1801) Lamarck (Jean Baptiste Pierre Antoine de Monet). Proposes a mechanism responsible for changes in species.
  - Env changes -> needs -> behavior -> body structures
  - Gradual adoptions to environment.
  - Germoplasm (cells with genetic information) genotype
  - Somatoplasm (cells with no genetic information) phenotype
  - There is more work on Evolution than time to present it!

EC Seminal Work
- (1957) Box – Evolutionary operation: A method for increasing industrial productivity
- (1958) Fredberg – A learning machine
- (1962) Holland – Outline for a logical theory of adaptive systems
- (1962) Bremermann – Optimization through evolution and recombination
- (1975) Holland – Adaptation in natural and artificial systems

Evolutionary Computation
- Benefits
  - Flexibility and adaptability
  - Robustness
  - Suited for complex optimization problems
- Main streams
  - Genetic Algorithms
  - Evolution Strategies
  - Evolutionary Programming

Evolution
- Nature-inspired Systems
  - E.g. Neuronal Networks
- Evolution through natural selection and mutation (Darwin, 1859)
- Plants and animals evolved from primitive species
- Genetic coding
- Survival of the fittest
- Alpha males
- Species survival

Evolution
- Takes place in long time (generaciones)
- Survival of the fittest
- The strongest mates
- Crossover, mutation, and selection
- Adaptation = process of progressive modification of general structures to adapt to the environment
- Meta-learning process at a generational level
Genetic Algorithms

- First adaptation models were introduced by John H. Holland in his book (1975).
  - "Adaptation in Natural and Artificial Systems"
- Genetic Algorithm → algorithmic techniques inspired on genetic principles, used to simulate evolutionary processes.

Terminology

- **DNA**: Deoxyribonucleic acid - contains the genetic instructions used in the development and functioning of all known living organisms
- **Nucleotides**: DNA consists of two long polymers of simple units called nucleotides ([G]uanine, [C]ytosine, [A]denine, and [T]hymine)
- **Chromosomes**: DNA + proteins
- **Genome**: the entirety of an organism’s hereditary information. Both the genes and the non-coding sequences of the DNA
- **Gene**: molecular unit of heredity of a living organism
- **Allele**: (allelomorph) of two or more forms of a gene
- **Gamete**: cell that fuses with another cell during fertilization.
- **Haploid**: The haploid number (n) is the number of chromosomes in a gamete.
- **Diploid**: cells with two homologous copies of each chromosome
- **Population**: all organisms that belong to the same species and live in the same geographical area
- **Individual**: a member of a population
- **Genotype**: genetic contents of an individual
- **Phenotype**: an organism’s observable characteristics or traits

Terminology

- **Gamete**: cell that fuses with another cell during fertilization.
- **Haploid**: The haploid number (n) is the number of chromosomes in a gamete.
- **Diploid**: cells with two homologous copies of each chromosome
- **Population**: all organisms that belong to the same species and live in the same geographical area
- **Individual**: a member of a population
- **Genotype**: genetic contents of an individual
- **Phenotype**: an organism’s observable characteristics or traits
- **Mutation**: accidental changes in a genomic sequence of DNA (copy mistakes)
- **Environment**: (biotic and abiotic) surrounding of an organism - factors that influence their survival, development and evolution
- **Fitness**: ability to survive and reproduce
- **Fitness function**: objective function that summarises how close an individual is to achieving the goal
- **Fitness Landscape**: hypersurface obtained applying the fitness function to each point of the search space

Terminology

- **Selection**: process by which some individuals from a population are chosen to reproduce → typically based on fitness
- **Hard selection**: Only the fittest individuals are chosen to produce offspring
- **Soft selection**: Probabilistic mechanism to include parents with relatively low fitnesses
- **Pleiotropy**: one gene influences multiple phenotypic traits
- **Polygene**: a group of non-allelic genes that together influence a phenotypic trait
- **Speciation**: evolutionary process by which new species arise
- **Genetic drift**: change in the frequency of a gene variant (allele) in a population due to randomness – takes many generations
- **Reproduction**: biological process by which new "offspring" are produced from their "parents"
  - **Sexual**
  - **Asexual**
- **Evolution**: change in the inherited characteristics of populations over successive generations
- **Coevolution**: change of a biological object triggered by the change of a related object (predator-prey, symbiosis, etc.)
Evolution: Basic Concepts

- Evolution = Two-level adaptation
- Adaptation = Progressive modification process leading to a better interaction with the environment

- AGs is an Evolutionary Algorithm
- Chromosome = Gene string

Evolution: Basic Concepts

- Genome → Genetic information
- Genome → Genetic variation
- Genetic changes → Mutation/Crossover
- Genome → Inheritance

Genetic Algorithms

- Original
- Crossover
- Mutation
Genetic Algorithms

Example
- Given a set of points in a (2D) coordinate system
- Determine the parabola that produced them
- Data contains noise

\((\{-10, 247.99\}, \{-9, 210.978\}, \{-8, 175.88\}, \{-7, 115.042\}, \ldots\})

Example

Evolution: Basic Concepts
- Genotype = DNA = Structure

\([-3, 4, 100]\)

- Phenotype = Characteristics

Experiment 1
- 20 individuals
- 5 offspring
- 10 generations

\([-3x^2 + 4x + 100]\)
Experiment 2
- 100 individuals
- 50 offspring
- 10 generations

Experiment 3
- 200 individuals
- 100 offspring
- 10 generations

AGs - Example
- Population convergence
- Generation 1

AGs - Example
- Population convergence
- Generation 2

AGs - Example
- Population convergence
- Generation 5

AGs - Example
- Population convergence
- Generation 8
AGs - Example

- Population convergence
- Generation 10

AGs - Example

- Convergence plot

Optimization

- Find $\hat{x} \in M$
- Such that $f(\hat{x}) \rightarrow \max$
- $f$ can be very complex

Global optimum

- Determine $\hat{x}^*$
- Such that $\forall x \in M : f(\hat{x}) \leq f(\hat{x}^*) = f^*$

Optimization

Multimodality:

- Several optima $\hat{x}^*$
- Such that $\exists \epsilon > 0 : \forall \hat{x} \in M : \rho(\hat{x}, \hat{x}^*) < \epsilon \Rightarrow f(\hat{x}) \leq f(\hat{x}^*)$
- $\rho$ - distance

Restrictions:

- $g_j : M \rightarrow \mathbb{R}$
- Feasible solutions $F \subseteq M$
- Such that $F = \{ x \in M | g_j(x) \geq 0 \ \forall j \}$

Optimization

Factors that make it harder:

- Dimensionality
- Non-linearity
- Non-differentiability
- Noise
- Dynamic functions

Conclusions

- $CE = $ Optimizers
- Can deal with systems:
  - Non-differentiable
  - Discontinuous
  - Hybrid
  - Restrictions (forbidden regions)
  - Curse of dimensionality
Conclusions

- Straightforward representations (not always)
- Restrictions – hard to deal with
- No recipe to know what Meta-heuristics to use
- Non deterministic
- Any-time algorithms
- Slow, but work
- Reduced total time