

---

**Ano Letivo** 2019-20

---

**Unidade Curricular** COMPUTAÇÃO EVOLUTIVA

---

**Cursos** ENGENHARIA INFORMÁTICA (2.º ciclo)  
ENGENHARIA ELETRÓNICA E TELECOMUNICAÇÕES (Mestrado Integrado) (\*)

(\*) Curso onde a unidade curricular é opcional

---

**Unidade Orgânica** Faculdade de Ciências e Tecnologia

---

**Código da Unidade Curricular** 14741034

---

**Área Científica** CIÊNCIA DE COMPUTADORES

---

**Sigla**

---

**Línguas de Aprendizagem** Inglês-EN

---

**Modalidade de ensino** Presencial

---

**Docente Responsável** Fernando Miguel Pais da Graça Lobo

---

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
Fernando Miguel Pais da Graça Lobo	T; TP	T1; TP1	30T; 30TP

\* Para turmas lecionadas conjuntamente, apenas é contabilizada a carga horária de uma delas.

ANO	PERÍODO DE FUNCIONAMENTO*	HORAS DE CONTACTO	HORAS TOTAIS DE TRABALHO	ECTS
1º	S1	30T; 30TP	168	6

\* A-Anual;S-Semestral;Q-Quadrimestral;T-Trimestral

#### Precedências

Sem precedências

#### Conhecimentos Prévios recomendados

Programação, Algoritmos e Estruturas de Dados, Probabilidades e Estatística.

#### Objetivos de aprendizagem (conhecimentos, aptidões e competências)

- Aprender os mecanismos de funcionamento das principais classes de Algoritmos Evolutivos (AEs)
- Ser capaz de implementar um AE e aplicá-lo num problema concreto.
- Desenvolver a qualidade de escrita técnica e de realização de apresentações.

#### Conteúdos programáticos

- O que é Computação Evolutiva? Perspectiva histórica. Principais classes de Algoritmos Evolutivos.
- O algoritmo genético simples. Principais métodos de selecção, recombinação, mutação, e reposição.
- Representações e concepção de operadores. Utilização de conhecimento específico.
- Estratégias evolutivas. A regra de 1/5. Auto-adaptação da força da mutação.
- Programação genética.
- Computação evolutiva interactiva.
- Tratamento de restrições.
- Optimização multi-modal.
- Optimização multi-objectivo.
- Teoria básica dos algoritmos genéticos. Limitações dos AEs simples. Dificuldade de problemas e o Teorema NFL.
- Decomposição de Goldberg para AGs competentes.
- Especificação de parâmetros em AEs. Avaliação de performance.
- Ideias chave de AEs baseados em modelos.

#### Demonstração da coerência dos conteúdos programáticos com os objetivos de aprendizagem da unidade curricular

A matéria abordada nos conteúdos programáticos irá capacitar os alunos com as competências necessárias para que os objectivos de aprendizagem enunciados sejam atingidos. Obviamente que para tal acontecer, espera-se que os alunos acompanhem a matéria ao longo do semestre e realizem os exercícios e trabalhos práticos propostos.

### Metodologias de ensino (avaliação incluída)

Nas aulas T, a matéria é apresentada e explicada aos estudantes, acompanhada de exemplos ilustrativos.

Nas aulas P os estudantes realizam exercícios de programação onde exercitam a matéria dada nas aulas teóricas e trabalham no projecto da disciplina.

Avaliação:

- exercícios de programação (10%)
- projecto (50%)
- exame final (40%)

O projecto é feito individualmente e consiste na aplicação de computação evolutiva num problema à escolha do estudante. Para a componente do projecto, o estudante terá de:

- escrever um relatório técnico (máximo de 10 páginas A4) descrevendo a aplicação de CE ao problema escolhido. (60%)
  - fazer uma apresentação oral de 20 minutos sobre o trabalho realizado. (20%)
  - ler e rever o relatório de um dos colegas. (10%)
  - submeter todo o código realizado na obtenção dos resultados apresentados no relatório técnico. (10%)
- 

### Demonstração da coerência das metodologias de ensino com os objetivos de aprendizagem da unidade curricular

Os objectivos de aprendizagem só serão alcançados se a matéria dada nas aulas teóricas for exercitada na prática. A metodologia de ensino adoptada e o método de avaliação proposto dão ênfase a este aspecto.

---

### Bibliografia principal

Não haverá um livro de texto único recomendado. Serão recomendados capítulos seleccionados dos seguintes livros, bem como artigos fornecidos pelo professor.

- Genetic Algorithms in Search, Optimization and Machine Learning, by David E. Goldberg, Addison-Wesley, 1989.
- Introduction to Evolutionary Computing, by A. E. Eiben and J. E. Smith, Springer, 2003.
- Evolutionary Computation 1: Basic Algorithms and Operators, edited by T. Baeck, D. B. Fogel, Z. Michalewick, 2000.
- Evolutionary Computation 2: Advanced Algorithms and Operators, edited by T. Baeck, D. B. Fogel, Z. Michalewick, 2000.
- Essentials of Metaheuristics (free at <http://www.cs.gmu.edu/~sean/book/metaheuristics/>), by Sean Luke, 2010.
- A Field Guide to Genetic Programming (free at <http://www.gp-field-guide.org.uk/>) by Riccardo Poli, William Langdon, and Nicholas McPhee.

**Academic Year** 2019-20

**Course unit** EVOLUTIVE COMPUTING

**Courses** INFORMATICS ENGINEERING  
ELECTRONIC ENGINEERING AND TELECOMMUNICATIONS (Integrated Master's) (\*)

(\*) Optional course unit for this course

**Faculty / School** FACULTY OF SCIENCES AND TECHNOLOGY

**Main Scientific Area** CIÊNCIA DE COMPUTADORES

**Acronym**

**Language of instruction** English-EN

**Teaching/Learning modality** In-person lectures.

**Coordinating teacher** Fernando Miguel Pais da Graça Lobo

Teaching staff	Type	Classes	Hours (*)
Fernando Miguel Pais da Graça Lobo	T; TP	T1; TP1	30T; 30TP

\* For classes taught jointly, it is only accounted the workload of one.

#### Contact hours

T	TP	PL	TC	S	E	OT	O	Total
30	30	0	0	0	0	0	0	168

T - Theoretical; TP - Theoretical and practical ; PL - Practical and laboratorial; TC - Field Work; S - Seminar; E - Training; OT - Tutorial; O - Other

#### Pre-requisites

no pre-requisites

#### Prior knowledge and skills

Computer Programming, Algorithms and Data Structures, Statistics and Probability.

#### The students intended learning outcomes (knowledge, skills and competences)

- Learn the working principles of the major classes of Evolutionary Algorithms (EAs)
- Be able to implement an EA and apply it to a particular domain
- Develop writing and presentation skills

#### Syllabus

- What is Evolutionary Computation? Historical perspective. Major classes of Evolutionary Algorithms. Local vs global search methods.
- Simple genetic algorithms. Major methods for selection, recombination, mutation, and replacement.
- Representations and design of operators. Using problem specific information.
- Evolution strategies. The 1/5 rule. Self-adaptation of mutation step sizes.
- Genetic programming.
- Interactive Evolutionary Computation.
- Constraint handling.
- Multimodal optimization.
- Multi-objective optimization.
- Basic GA theory. Limitations of simple EAs. Problem difficulty and the NFL theorem.
- Goldberg's decomposition for competent GAs.
- Parameter setting in EAs. Performance assessment.
- Key ideas of Model based EAs.

#### Demonstration of the syllabus coherence with the curricular unit's learning objectives

The topics covered in the syllabus allow the students to acquire the necessary skills to reach the learning outcomes. It is however expected that students follow the material throughout the semester and do the proposed practical assignments and the course practical project.

### Teaching methodologies (including evaluation)

In the main T lectures, the course materials are explained to students along with illustrative examples.  
In the P lectures students have hands-on experience on the course materials and work on their projects.

Grading:

- programming assignments (10%)
- project (50%)
- final exam (40%)

The project is done individually and consists of applying evolutionary computation to an application domain of the student's choice. For the project component, students will have to:

- write a technical report (10 A4 pages maximum) describing your EC application. (60%)
  - make a 20-minute oral presentation of your work. (20%)
  - read and carefully review a report from your classmates. (10%)
  - submit all the code that you did to obtain the results shown in your technical report. (10%)
- 

### Demonstration of the coherence between the teaching methodologies and the learning outcomes

The learning outcomes will only be reached if the material covered in the main T lectures is exercised in practice. The teaching methodology and the proposed evaluation/grading give emphasis to this aspect.

---

### Main Bibliography

No particular textbook will be followed. Selected chapters from the following books will be recommended, along with research papers provided by the instructor.

- Genetic Algorithms in Search, Optimization and Machine Learning, by David E. Goldberg, Addison-Wesley, 1989.
- Introduction to Evolutionary Computing, by A. E. Eiben and J. E. Smith, Springer, 2003.
- Evolutionary Computation 1: Basic Algorithms and Operators, edited by T. Baeck, D. B. Fogel, Z. Michalewick, 2000.
- Evolutionary Computation 2: Advanced Algorithms and Operators, edited by T. Baeck, D. B. Fogel, Z. Michalewick, 2000.
- Essentials of Metaheuristics (free at <http://www.cs.gmu.edu/~sean/book/metaheuristics/>), by Sean Luke, 2010.
- A Field Guide to Genetic Programming (free at <http://www.gp-field-guide.org.uk/>) by Riccardo Poli, William Langdon, and Nicholas McPhee.