
Ano Letivo 2020-21

Unidade Curricular METAHEURÍSTICAS

Cursos ENGENHARIA INFORMÁTICA (2.º ciclo)

Unidade Orgânica Faculdade de Ciências e Tecnologia

Código da Unidade Curricular 14741065

Área Científica CIÊNCIA DE COMPUTADORES

Sigla

Línguas de Aprendizagem

Inglês-EN

Português-PT

Modalidade de ensino

Presencial ou à distância

Docente Responsável João Miguel Gago Pontes de Brito Lima

DOCENTE	TIPO DE AULA	TURMAS	TOTAL HORAS DE CONTACTO (*)
João Miguel Gago Pontes de Brito Lima	T; TP	T1; TP1	28T; 28TP

* 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	28T; 28TP	156	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.
- Tomar conhecimento de outras meta-heurísticas.
- 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 algoritmos 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.
- Referencia breve a outras meta-heurísticas

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:

- projecto (50%)
- exame final (50%)

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%)

Bibliografia principal

- 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. Michalewicz, 2000.
- Evolutionary Computation 2: Advanced Algorithms and Operators, edited by T. Baeck, D. B. Fogel, Z. Michalewicz, 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 2020-21

Course unit

Courses INFORMATICS ENGINEERING

Faculty / School FACULTY OF SCIENCES AND TECHNOLOGY

Main Scientific Area

Acronym

Language of instruction
Portuguese- PT
English-EN

Teaching/Learning modality
In-person or distance lectures

Coordinating teacher João Miguel Gago Pontes de Brito Lima

Teaching staff	Type	Classes	Hours (*)
João Miguel Gago Pontes de Brito Lima	T; TP	T1; TP1	28T; 28TP

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

Contact hours

T	TP	PL	TC	S	E	OT	O	Total
28	28	0	0	0	0	0	0	156

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
- Introduction to other metaheuristics.
- 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.
- Brief reference to other metaheuristics.

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:

- project (50%)
- final exam (50%)

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%)

Main Bibliography

- 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. Michalewicz, 2000.
- Evolutionary Computation 2: Advanced Algorithms and Operators, edited by T. Baeck, D. B. Fogel, Z. Michalewicz, 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.