250406 - ENGCOMP - Computational Engineering

**Degree competences to which the subject contributes**

**Specific:**
8200. The ability to apply knowledge of soil and rock mechanics to the study, design, construction and operation of foundations, cuts, fills, tunnels and other constructions over or through land, whatever its nature and state, and whatever the purpose of the work.
8228. Knowledge of and competence in the application of advanced structural design and calculations for structural analysis, based on knowledge and understanding of forces and their application to civil engineering structures. The ability to assess structural integrity.
8230. The ability to plan, dimension, construct and maintain hydraulic works.
8231. The ability to plan, evaluate and regulate the use of surface water and groundwater resources.
8233. Knowledge of and the ability to understand dynamic phenomena of the coastal ocean and atmosphere and respond to problems encountered in port and coastal areas, including the environmental impact of coastal interventions. The ability to analyse and plan maritime works.
8234. Knowledge of transport engineering and planning, transport types and functions, urban transport, management of public transport services, demand, costs, logistics, and financing of transport infrastructure and services.

**Teaching methodology**

Taught module delivery: fifteen weeks of teaching, coursework and self-study. Apart from the 4 hours per week in the classroom, self-study must last an average of 6 hours per week.

**Learning objectives of the subject**

Students will learn to design computational models for the mechanics of continuous media and for solving diagnostic problems encountered in engineering.

Upon completion of the course, students will be able to:
Develop computational models based on mechanics of continuous media and apply them to different areas of civil engineering, including soil and rock mechanics, structural analysis, hydrology and water resources, ports and coastal systems;
Develop discrete computational models and use them for network design in different areas of civil engineering, in particular transport, logistics, power distribution and infrastructure mapping;
Apply the uncertainty principle to data on the external actions and internal properties of systems;
Apply stochastic computational models and subject the results to statistical processing;
Use the results of computational models as the basis for design, analysis, optimisation and decision-making in civil engineering.

Computational engineering techniques for the modelling and solution of continuous equilibrium and evolution problems; Application to structural engineering, geotechnical engineering, transport engineering, maritime engineering and environmental engineering; Continuous optimisation techniques (linear programming and nonlinear programming): Application to optimal design, parameter identification and resource allocation; Discrete optimisation and combinatorial optimisation techniques: Application to network design; Monte Carlo simulation: Application to decision-making in management and planning

### Study load

<table>
<thead>
<tr>
<th>Total learning time: 150h</th>
<th>Theory classes: 17h 24m 11.60%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Practical classes: 8h 24m 5.60%</td>
</tr>
<tr>
<td></td>
<td>Laboratory classes: 26h 13.2m 17.48%</td>
</tr>
<tr>
<td></td>
<td>Guided activities: 1h 58,8m 1.32%</td>
</tr>
<tr>
<td></td>
<td>Self study: 96h 64.00%</td>
</tr>
</tbody>
</table>

Last update: 04-11-2015
1.- Modelling with ODEs

**Description:**
Initial value problems in numerical engineering
Mètodes simple step (Runge-Kutta) for initial value problems
Boundary value problems in engineering
The shooting method for boundary value problems
Solution of two problems (fields: structural analysis / geotechnics)
Adaptive numerical techniques
Variable step: step adaptation based on error control
The Runge-Kutta-Fehlberg 45 method (RKF45)
Solution of a case study (field: structural analysis)

**Specific objectives:**
To be able to model engineering problems with ODEs
To know and to use, with the help of Matlab or Octave, numerical techniques for initial value problems
To identify whether an engineering problem is an initial value problem or a boundary value problem
To know the strategy to solve boundary value problems, based on iterations (shoots) of an associated initial value problem
To model two case studies as boundary value problems: ODEs and boundary conditions
To sketch the solution of the boundary value problems with the shooting method
To be able to assess and control the quality of the numerical solution
To know and to be able to apply the RKF45 method to initial value and boundary value problems
To solve a boundary value problem with the RKF45 method (command ode45 in Matlab)
To be able to choose the accuracy needed in a computation depending on the output of interest
2.- Modelling with PDEs

Learning time: 43h 12m
- Theory classes: 8h
- Practical classes: 4h
- Laboratory classes: 6h
- Self study: 25h 12m

Description:
- Solution of two problems (field: hydraulics / soil mechanics)
- Solution of a case study (field: structural analysis) with the FEM
- Eigenvalue problems in engineering
- Wave equation (vibrations, sea waves, sound). Eigenmodes and eigenfrequencies
- Standard and generalized eigenvalue problems
- Properties of the symmetric eigenvalue problem
- Methods of direct vector iteration and inverse vector iteration
- Solution of two case studies (field: structural dynamics / marine engineering)
- Diffusion and convection-diffusion problems in engineering
- Solution of evolution problems with the FEM

Solution of a case study (field: environmental engineering)
Nonlinear equilibrium problems in engineering
Incremental-iterative strategy: the Newton-Raphson method
Solution of a case study (field: structural analysis) with the FEM

Specific objectives:
- To model two case studies as equilibrium problems: PDEs and boundary conditions
- To sketch the solution of equilibrium problems with the finite element method (FEM)
- To know the internal organization of a finite element code
- To be able to solve equilibrium problems with a finite element code
- To interpret results, to evaluate and control the quality of outputs of interest
- To know the most relevant eigenvalue problems in engineering
- To give a physical interpretation to eigenvectors (modes) and eigenvalues (frequencies)
- To know numerical techniques to determine the eigenvalues and eigenvectors of interest
- To be able to determine eigenfrequencies and eigenmodes of interest
- To understand the basic idea of modal analysis
- To identify whether a problem is an equilibrium or an evolution problem
- To know numerical techniques to treat time variation and convection
- To model a case study as a convection-diffusion problem
- To sketch its solution with the FEM
- To identify the different sources of nonlinearity in engineering
- To know numerical FEM techniques for nonlinear problems
- To model nonlinearity in a case study
- To be able to solve nonlinear problems with a finite element code
- To understand the large difference in complexity between linear and nonlinear problems

Test #1

Learning time: 4h 48m
- Laboratory classes: 2h
- Self study: 2h 48m
| **Test #2** | **Learning time:** 4h 48m  
 | | Laboratory classes: 2h  
 | | Self study : 2h 48m |
3.- Optimization and simulation

Learning time: 40h 48m
  Theory classes: 14h
  Practical classes: 2h
  Laboratory classes: 1h
  Self study: 23h 48m

Description:
Introduction to optimization.
Types of problems in engineering

Numerical techniques for continuous optimization problems without constraints
Models in engineering with unknown parameters
Nonlinear least-squares fitting from experimental measures
Numerical techniques: Newton, Levenberg-Marquardt
Heuristic techniques: genetic algorithms
Solution of a case study (field: soil mechanics / structural analysis)
Optimization problems in engineering
Equality and inequality constraints
Optimization with equality constraints: Lagrange multipliers
Optimization with inequality constraints: active constraints, barrier functions, penalty functions
Linear programming problems in engineering
Linear goal function and linear constraints
The simplex method
Solution of a case study (field: resource allocation)
Network problems in engineering
Exact and heuristic algorithms in discrete and combinatorial optimization
Simulation: Montecarlo method

Specific objectives:
Identify the different types of engineering problems: direct, optimal design, optimal identification and optimal control.
Establish the relationship between unconstrained minimization and the solution of nonlinear systems of equations

To discuss whether a model is linear or nonlinear in its parameters
To formulate parameter identification as a minimization problem To know how to apply Levenberg-Marquardt method for parameter fitting
To know the basic idea of genetic algorithms
To model a case study as a problem of parameter identification
To solve it with the Levenberg-Marquardt method
To know the different types of optimization problems in engineering
To understand the role played by the constraints
To understand the different numerical treatment of problems with and without constraints
To know the basic idea of numerical techniques for constraints
To know different types of linear programming problems in engineering
To understand the basic rules of the simplex method
To model a resource allocation case study as a linear programming problem
To solve it with the simplex method
To formulate network design problems as discrete /combinatorial optimization problems
To know the basic idea of the two types of techniques: exact and approximate
Decision making
## Qualification system

1. The course assignment must be submitted on the announced due date. Otherwise, a failed mark will be considered.
2. The tests will cover all the taught material from the beginning of the module.
3. Students must bring a calculator with no internet connection to the tests. Mobile phones, computers, tablets, or any other electronic devices are forbidden during tests.
4. Notes, textbooks, solved problems, or any other documents are forbidden during tests.
5. Academic dishonesty will be severely punished in accordance with current academic regulations.
6. The final mark for the module is obtained from every rated activity according to:

   \[
   \text{Mark} = A^{(1/2)} \times TC^{(1/4)} \times (E + TD)^{(1/4)}
   \]

   If Mark < 5, \( \text{Mark} = \max(\text{Mark}, PC^{(3/4)} \times TC^{(1/4)}) \)

   \( A \) is the arithmetic mean of the tests: \( A = (A1 + A2 + A3)/3 \)

   \( TC \) is the arithmetic mean of course assignment: 50% for the poster and its presentation, and 50% for the final report.

   The course contains various assignments in class (E) and tutorials in the computer room (TD). Some of them will be submitted at the end of the class for grading. Of these, the best marks (three out of each four) will be used to compute the arithmetic mean of assignments in class and tutorials, \( E + TD \).

   \( PC \) is the mark of a global assessment.

## Regulations for carrying out activities

Failure to perform a laboratory or continuous assessment activity in the scheduled period will result in a mark of zero in that activity.

The Course Assignment is required in order to be marked.
Bibliography

**Basic:**


**Complementary:**