230481 - COMBIO - Computational Biophysics

Coordinating unit: 230 - ETSETB - Barcelona School of Telecommunications Engineering
Teaching unit: 748 - FIS - Department of Physics
Academic year: 2019
Degree: BACHELOR'S DEGREE IN ENGINEERING PHYSICS (Syllabus 2011). (Teaching unit Optional)
ECTS credits: 6
Teaching languages: English

Teaching staff

Coordinator: Alvarez Lacalle, Enrique
Others: Pons Rivero, Antonio Javier
Pastor Satorras, Romualdo
Alonso Muñoz, Sergio

Opening hours

Timetable: Students will be informed using atenea's webpage

Prior skills

- Basic knowledge of the human body obtained in ESO and Biophysics1.
- Knowledge of matlab or any other programming language (seen in MNC1)
- Knowledge of the main step/multistep algorithm to resolve ordinary differential equations, specially Runge-Kutta methods. (seen in MNC2)
- Knowledge and familiarity with the main concepts of ordinary differential equations and partial differential equations. (seen in MM2)
- Knowledge and familiarity with the basic concepts of bacteria, cell and the central dogma of biology (seen in BIOF2)
- Familiarity with the concepts of analytical mechanics related with phase space and dynamical attractors. (seen in MECF)

Requirements

Numerical and Computational Methods 2
Mecànica
Mètodes matemàtiques 2
Biophysics 2

Degree competences to which the subject contributes

Specific:
INF1. Understanding and mastery of computer programming, use of operative systems and computational tools (scientific software). Skills to implement numerical algorithms in languages of low (C, F90) and high (Matlab) level. 
BIOC1. Ability to describe in general the structure of living things, from cellular to systemic level. Ability to analyze the constraints imposed by the physics laws to the development of biological systems, and the biological solutions to engineering problems.
BIOC2. Ability to analyze biological systems as complex systems.

INF2. Ability to solve problems in physics and engineering using fundamental numerical methods: experimental data processing, interpolation, roots of nonlinear equations, numerical linear algebra and optimization, quadrature and integration of differential equations, properly weighting their different aspects (accuracy, stability and efficiency or cost).

Generical:
Once the student has finished the course he/she must be able to:

- Describe the four main methods of addressing spatio-temporal dynamics in extended systems used in physics: Compartmental models, iterative models, spatio-temporal models and network methods. This includes discerning which one is optimal for addressing the problem.

- Implement a random number generator in any given numerical simulation and know the Gillespie algorithm to code stochastic variables in time propagators. Implement extended nonlinear simulations with diffusion processes and different boundary conditions. Understand the concept of biological network, how to implement it numerically on a computer and how to obtain useful topological information of it. Learn how to implement realistic dynamical processes on top of networks.

- Define the basic dynamical interaction in a typical metabolic process and the kinetic reaction and network theory used to describe it. It must also be able to code a typical oscillatory genetic network and a typical signaling process. It must be able to describe briefly the general gene-protein network structure of the cell. It must be able to analyze the possible stochastic elements in a given biological system and implement it dynamics.

Transversal:

1. THIRD LANGUAGE. Learning a third language, preferably English, to a degree of oral and written fluency that fits in with the future needs of the graduates of each course.
2. SELF-DIRECTED LEARNING - Level 3. Applying the knowledge gained in completing a task according to its relevance and importance. Deciding how to carry out a task, the amount of time to be devoted to it and the most suitable information sources.

Teaching methodology

The course will be centered in code-implementation classes and its theoretical and biological interpretation. Besides the normal classes, a few seminars showing examples of different types of modeling in biology will be given. During the course there will be basically two types of classes

- Theoretical classes will review and address definitions and frameworks in the area of computational sciences, math and biophysics. The basic structure of the class will be to expose the different concepts using blackboard, videos, and interactive resources.

- Practical classes will include time for the student to write and develop small codes. These codes will be useful both as a numerical examples and as toy models of biophysical problems. One single class can be divided in theoretical and practical time if the topic at hand requires it.

Besides regular classes two different methodologies will be used:
- During the first week a special training program to code in python will be developed using the previous knowledge the students have in matlab.
- Seminar series will be arranged during the course. Professors will give seminars overviewing topics of computational biophysics which aren't addressed in the program so that students can overview more complex numerical methods and analysis.

All the resources will be published in the ATENEA website.

Learning objectives of the subject

Once the student has finished the course he/she must be able to:

- Describe the four main methods of addressing spatio-temporal dynamics in extended systems used in physics: Compartmental models, iterative models, spatio-temporal models and network methods. This includes discerning which one is optimal for addressing the problem.

- Implement a random number generator in any given numerical simulation and know the Gillespie algorithm to code stochastic variables in time propagators. Implement extended nonlinear simulations with diffusion processes and different boundary conditions. Understand the concept of biological network, how to implement it numerically on a computer and how to obtain useful topological information of it. Learn how to implement realistic dynamical processes on top of networks.

- Define the basic dynamical interaction in a typical metabolic process and the kinetic reaction and network theory used to describe it. It must also be able to code a typical oscillatory genetic network and a typical signaling process. It must be able to describe briefly the general gene-protein network structure of the cell. It must be able to analyze the possible stochastic elements in a given biological system and implement it dynamics.
Make linear stability analysis of pattern forming systems. Be able to solve linear and nonlinear initial value and boundary value problems using different numerical implementations analyzing its stability and accuracy. Understanding the dynamical features of reaction-diffusion-advection systems and to give different examples of pattern forming systems in Biology.

-Understand simple models of population dynamics and its relation to ecology, learning the different factors that can affect the growth and development of a population: spatio-temporal effects, effects of generational delays. At the same time, learn how to control possible chaotic effects in non-linear ecological models.

-Define the basic aspects of epidemiology and the relevant models that display the main features of realistic diseases. Understand the effects of a complex pattern of physical contacts in the spread of a biological disease.

Regarding the seminar series, students must be able to summarize any seminar presentation and be able to explain briefly three areas where biophysics approaches are presently top-edge.

Finally, students must perform a project where they develop a code to show expertise in one particular biophysics problem. At the end of this project, students must be able to show that they can understand the physiology or structure of a given biophysical problem in order to properly simulate it. They must be able to structure a code and clarify its different subroutines checking that the code works correctly and, if possible, perform simulations to draw conclusions about the problem or about future improvements of the code.

**Study load**

<table>
<thead>
<tr>
<th>Total learning time: 150h</th>
<th>Hours large group:</th>
<th>65h</th>
<th>43.33%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Self study:</td>
<td>85h</td>
<td></td>
<td>56.67%</td>
</tr>
</tbody>
</table>
### Introduction

**Description:**
- System biology. Introduction and general approach.

- Introduction to python language: basic commands.

- Basic data processing in python.
  - Managing of data sets to obtain relevant statistical descriptions. Reading and writing numerical data. Analysis, linear and non-linear regression of empirical data. Efficient graphical representation.

**Related activities:**
Presentación AV1

---

### Non-extended dynamical systems. Genetic networks.

**Learning time:** 36h
- Theory classes: 14h
- Practical classes: 6h
- Self study: 16h

**Description:**
- Introduction to ODE’s. Fixed points and nullclines.
  - High order differential equations and ODE’s. Fixed points in one-dimensional systems. Generalization. Oscillation and periodicity in two-dimensional system. Nullclines in two-dimensional systems.

- Slaving and freezing conditions. Applications to cell reactions.
  - Simulation of different time scales. Long time scales and freezing variables. Fast time scales and slaving. Optimal time steps in Euler methods.


- Stochastic modeling: Random number generator. Application to cell signaling.
### Spatially extended systems. Pattern formation in biology

**Learning time:** 36h  
- **Theory classes:** 14h  
- **Practical classes:** 6h  
- **Self study:** 16h

**Description:**
- Introduction to Pattern formation  
  - Linear Stability analysis. Amplitude Equations.
- Reaction-diffusion systems.  
- Reaction-diffusion-advection systems.  
- Pseudo-Spectral methods in biophysics.  
- Patterns in the Brain.  
  - Neural fields. Patterns not derived from conservation equations. Derivation and biological interpretation.


**Learning time:** 28h  
- **Theory classes:** 10h  
- **Practical classes:** 5h  
- **Self study:** 13h

**Description:**
- Introduction to networks and their implementation.  
  - Definition of a network. Efficient numerical implementation on the computer using higher order data structures. Topological observables and description of efficient algorithms to compute them.
- SIS models in networks. Application to epidemiology.  
Planning of activities

**Seminar series**

<table>
<thead>
<tr>
<th>Learning time: 5h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Practical classes: 5h</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
</tr>
<tr>
<td>Modeling spatial population genetics.</td>
</tr>
<tr>
<td>Modeling self-organization in biology.</td>
</tr>
<tr>
<td>Modeling infectious diseases.</td>
</tr>
</tbody>
</table>

**AV1: Main project: Numerical simulation of a biophysical system.**

<table>
<thead>
<tr>
<th>Hours: 43h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guided activities: 3h</td>
</tr>
<tr>
<td>Self study: 40h</td>
</tr>
<tr>
<td><strong>Description:</strong></td>
</tr>
<tr>
<td>This project consists on developing a simulation of a relatively complex biophysical system. Students will work in pairs (not necessarily) and learn the fundamentals of the biophysical system, will decide how to address a simulation of the system and will use the numerical algorithms learnt in class (or in autonomous learning) to implement it. The code written by the students will have one of two aims: simulate numerically a full biophysical system or understand via simulations a particular feature of the same. The student will have to test the code modularly. Once the code is up and running, they will comment and comment it and proceed to use it to attain the desired goal. The project will end with a short report of the work done according to given guidelines.</td>
</tr>
<tr>
<td><strong>Support materials:</strong></td>
</tr>
<tr>
<td>The general structure of the project will be given to the student together with bibliographic support and notes if necessary. The main structure of the code will be developed with the support of a tutor if necessary.</td>
</tr>
<tr>
<td><strong>Descriptions of the assignments due and their relation to the assessment:</strong></td>
</tr>
<tr>
<td>Commented and documented code (up to 500 lines of code +100 in comments with one page documentation).</td>
</tr>
<tr>
<td>Standard brief report with abstract, introduction, implemented model, code tests and results with conclusion (maximum 4 pages).</td>
</tr>
<tr>
<td>Evaluation 40% of the total grade</td>
</tr>
<tr>
<td><strong>Specific objectives:</strong></td>
</tr>
<tr>
<td>Students should develop a project with the aim of applying the different numerical techniques and theoretical knowledge explained during the course. When necessary, students may need to study a new numerical technique specifically useful for the project. All this means the student must learn autonomously, plan and focus in order to reach a clear goal. This would allow the students to consolidate the knowledge of the course, develop the ability to solve new problems and communicate this knowledge clearly and efficiently.</td>
</tr>
</tbody>
</table>

**Qualification system**

The students' evaluation will consist on grading the work done in class and at home through handed-in homework (HE), and the formal evaluation of the project (PE). There will not be a mid-term exam nor a final exam. The final mark will be given by:

0.6*HE + 0.4*PE
230481 - COMBIO - Computational Biophysics

Regulations for carrying out activities

All activities are compulsory. Any report or project which is not delivered or presented will have a grade of zero.

Bibliography

Basic:


Complementary:


Others resources:

Hyperlink

The virtual heart

http://thevirtualheart.org/

Human Brain Project

https://www.humanbrainproject.eu/es/discover/the-project/overview

Computer material

Curs a atenea.

Web resources in atenea