Modelling and simulation - course description

General information

Course name	Modelling and simulation
Course ID	11.9-WE-AutP-ModSymul-Er
Faculty	Faculty of Computer Science, Electrical Engineering and Automatics
Field of study	Automatic Control and Robotics
Education profile	academic
Level of studies	First-cycle Erasmus programme
Beginning semester	winter term 2019/2020

Course information

Semester	2
ECTS credits to win	6
Course type	obligatory
Teaching language	english
Author of syllabus	• prof. dr hab. inż. Dariusz Uciński

Classes forms

The class form	Hours per semester (full-time)	Hours per week (full-time)	Hours per semester (part-time)	Hours per week (part-time)	Form of assignment
Lecture	30	2	-	-	Exam
Laboratory	30	2	-	-	Credit with grade

Aim of the course

- To provide students with methods, techniques and tools for modeling and simulation of continuous-time and discrete-time systems.
- To introduce students to mathematical models of typical electromechanical systems.
- To use Matlab / Octave / Scilab and Maple / Maxima for solving common engineering problems.

Prerequisites

Mathematical Analysis, Linear Algebra with Analytic Geometry

Scope

Introduction to Maple V and Maxima. Elements of the language. Assignment. Basic types: sequences, sets, lists, tables, arrays and strings. Calling procedures. Using apostrophes. Internal data representation. Solving linear and nonlinear equations. Functions for linear algebra and mathematical analysis. Simplification of expressions: simplify, factor, expand, convert, normal, combine, map i assume. 2D and 3D graphics. Programming foundations. Applications in mathematical analysis, linear algebra, statistics and selected engineering problems.

Mathematical models of dynamic systems. Models, modelling and simulation. Classification of modelling methods. Goals and stages of modelling. Basic physical laws. Exemplary models of mechanical, electrical, economical and control systems.

Ordinary differential equations. Definitions, classification. Examples of geometric and physical problems leading to differential equations. Geometrical interpretation. Direction field. Integrals of ordinary differential equations. Existence and uniqueness of solutions. First-order equations in normal form. Equations with separated variables. Homogeneous equations. Linear equations. Bernoulli and Riccati equations. Complete differential equations. Trajectories. N-th order linear differential equations. General integrals of linear equations. Fundamental matrix and its properties. Second-order equations with variable coefficients. Systems of nonlinear ordinary differential equations.

Numerical methods of solving ordinary differentia equations. One-step methods: Euler method, trapezoid method (Crank-Nicolson), Heun method. Explicit and implicit schemes. Multistep methods: Adams methods, backward difference methods. Predictor-corrector methods. Runge-Kutta methods. Adaptive step size selection. Systems of ordinary differential equations. Stiff problems.

Continuous linear dynamic systems. Descriptions: ordinary differential equations, transfer functions. Determining responses to any inputs. Matrix transfer functions. Examples of fundamental elements. State equations of linear systems.

Discrete linear dynamic systems. Engineering examples. Difference equations. Transfer functions of discrete systems. State equations.

Matlab-Simulink and Scilab-Xcos environments. Characteristics and applications. Operations on vectors and matrices. Logical expressions. Basic mathematical functions. 2D and 3D graphics. Animation. Low-level graphical functions. Iteration instructions. Scripts and functions. Elements of programming. Debugger. Code efficiency. Recursion. Vectorization of algorithms. Operating on strings. Nonstandard data structures: sparse matrices, structures, cell arrays, multidimensional arrays. Building graphical user interfaces. Operations on files. Calling MATLAB from C programs. Selected toolboxes. Building models of continuous and discrete processes. Simulink: blocks, S-functions.

Building mathematical models based on the principle of least action. Models of mechanical systems. Models of electrical systems. Models of electromechanical systems. Models of gases and liquids. Models of thermal systems. Models of chemical and biochemical processes. Model linearization. Implementation in MATLAB/Simulink.

Teaching methods

Lecture, laboratory exercises

Learning outcomes and methods of theirs verification

Outcome description	Outcome symbols	Methods of verification	The class form
Can identify and mathematically describe basic types of signals		• an exam - oral, descriptive, test and other • exam	• Lecture
Is aware of the importance of an adequate mathematical model in the system analysis and synthesis of control systems		• an exam - oral, descriptive, test and other • exam	• Lecture
Can build and simulate a mathematical model of a simple engineering electro-mechanical system		• an ongoing monitoring during classes • carrying out laboratory reports • on-going assessment of teaching laboratory activities, writing laboratory reports	• Laboratory
Can characterize the most important numerical methods of solving ordinary differential equations and implement them in a chosen programming language.		• an exam - oral, descriptive, test and other • an ongoing monitoring during classes • on-going assessment of teaching laboratory activities, test	• Lecture • Laboratory
Can enumerate and characterize basic models of mathematical dynamic systems		• an exam - oral, descriptive, test and other • exam	• Lecture
Can solve linear differential equations using basic analytical methods (method of separation of variables and variation of constant).		• an ongoing monitoring during classes • carrying out laboratory reports • on-going assessment of teaching laboratory activities, writing laboratory reports	• Laboratory
Can skilfully use numerical and symbolic modern environments for modeling and visualization of results (Matlab/Octave/Simulink, Scilab/Xcos, Maple/Maxima).		• an ongoing monitoring during classes • carrying out laboratory reports • ong-going assessment of teaching laboratory activities, writing laboratory reports	• Laboratory

Assignment conditions

Lecture – the main condition to get a pass is a sufficient mark in a written or oral exam.
Laboratory – the passing condition is to obtain positive marks from all laboratory exercises to be planned during the semester.
Calculation of the final grade: lecture 50% + laboratory 50%

Recommended reading

1. Christian Constanda (2017): Differential Equations: A Primer for Scientists and Engineers, Springer
2. Steven I. Gordon and Brian Guilfoos (2017): Introduction to Modeling and Simulation with MATLAB and Python, CRC Press
3. Carlos A. Smith and Scott W. Campbell (2016): A First Course in Differential Equations, Modeling, and Simulation 2nd Edition, CRC Press

Further reading

Notes

Modified by dr hab. inż. Wojciech Paszke, prof. UZ (last modification: 29-04-2020 09:22)