Ph.D. Program in Information Engineering

Course Catalogue

A.Y. 2019/2020

Rev. 1.3 – 8/11/2019

Revision History

Reference version: 1.0 - 15/10/2019

Rev. 1.1 – 23/10/2019

- Added number of credits (5) to the syllabus of course "*Advanced topics in scientific and parallel programming with practical application to the CAPRI HPC infrastructure*".

Rev. 1.2 – 24/10/2019

- Added number of credits (5) to the syllabus of course "*Control of Multivariable Systems: A Geometric Approach*".
- Summary: course "*Real-Time Systems and Applications*" moved from "*Control Theory and Applications*" to "*Computer Science*" cathegory.

Rev. 1.3 – 8/11/2019

- Course "*IE 15. Modeling and Simulation of Complex & Multi-Disciplinary Dynamical Systems*": added link to project abstracts and videos by students who took this class at MIT.
- Course "*IE 18. Applied Linear Algebra*": updated syllabus.

[back to **Summary**]

Summary

General Information

Courses - Transversal Skills Area

Courses - Information Engineering Area

Mathematical and Statistical Methods

Bioengineering

Electronics

Telecommunications

Control Theory and Applications

Computer Science

Course Index by Instructor

Coursework Requirements

The following requirements are valid for Ph.D. Students starting in October 2019 (35° cycle). In summary, Students shall **take courses for a minimum of 20 credits** and shall **attend a number of seminars**, following the rules detailed below.

Definitions

A **course** is a series of lectures given by an instructor (professor or university researcher), possibly accompanied by laboratory sessions, that includes an assessment of the student knowledge (final exam, graded homework or project, etc.). A course gives credits.

A **seminar**, in the context of the Ph.D. Program, is typically a talk, or a series of talks, given by an academic researcher or an eminent professional, that does not include an assessment. A seminar does not give credits.

Course requirements

- Take Ph.D. courses for **a minimum of 20 credits** by the end of the second year.
- **At least 10** of the above 20 credits shall be earned by the end of the first year.

Specific constraints to earn the minimum of 20 credits of courses:

- **Transversal Skills Area (TSK)**: at least 5 credits should come from courses belonging to the Transversal Skills area (labeled **TSK** in the course Summary).
- **Information Engineering Area (IE)**: students shall earn at least 10 credits by taking courses belonging to the Information Engineering Area (labeled **IE** in the course Summary).
- **External Courses**: up to a maximum of 5 credits may be earned by taking external courses (i.e. courses not included in this catalogue) falling in the following categories:
  - Courses appearing in the list of external courses approved by the Executive Board. The list of credited external courses is available on the Ph.D. Program main website.
  - Additional external courses might be included into the list after submission of a written request signed by the Student and his/her Supervisor. Only courses including an exam with grading are considered.
  - Courses from other Ph.D. School catalogues (provided they include a final exam with grading).
  - In order to get credit recognition for external courses, students shall obtain a certificate stating that the student attended the course and successfully passed the exam. Alternatively, the student may fill a Certification of Attendance with the course data and have it signed by the course instructor.

Seminar requirements

- Attend at least three of the **seminars/MOOCs** promoted by the Ph.D. Program (find the list on the website) by the end of the third year.

[back to **Summary**]

- Attend all the lectures of the **Distinguished Lecturer Series** program offered during the three-year Ph.D. course.
- Attend at least two modules of the **PhD Educational Week on Transferable Skills 2020 or 2021.**

Each first-year student must fill a tentative program of study form and upload it using the following link:

within October 31st. The program of study may be subsequently modified by submitting a new form no later than June 30th of the second year.

Students are expected to attend classes regularly. Punctuality is expected both from instructors and students. Instructors have to report to the Coordinator of the Ph.D. Program students missing classes without proper excuse.

Instructors shall complete student grading within 30 days after the end of lectures and, anyway, no later than August 31st.

Class Schedule

The class schedule is embedded in the Ph.D. Program Calendar. If you have a Google account, you may visualize the class schedule through the following link:

Class Schedule of 2019/20 PhD Courses for Google Calendar

You may also visualize the class schedule using any browser through the following link:

Class Schedule of 2019/20 PhD Courses

Most classes meet in DEI/D meeting room, located at the 1st floor of DEI/D building, location (4) in the map below, or in Room 318 DEI/G, located at the 3rd floor of DEI/G Building, location (3) in the map below, at the Dept. of Information Engineering, via Gradenigo 6/A, Padova.

Please, always check the class schedule in the calendar to verify the class meeting room.

In order to locate the rooms, you may find helpful the map of the Department buildings:

Map of the Department of Information Engineering

TSK 1. Entrepreneurship and Technology-based Startups

Course Area: Transversal Skills

Credits: 5

Instructors: Prof. Moreno Muffatto, Ing. Francesco Ferrati, Dipartimento di Ingegneria Industriale, Università di Padova

e-mail: moreno.muffatto@unipd.it, francesco.ferrati@unipd.it

Aim:

Topics:

From the idea to the market

- From a research project to an entrepreneurial project
- Recognize and evaluate an entrepreneurial opportunity
- Market dimension, customers profiles and value proposition
- Development of the product/service concept
- Go-to-Market strategies

Intellectual Property Rights

- Types of IPR (patent, copyright, trademark)
- The structure of a patent application (description, claims, etc)
- Getting a patent: the patenting process (step by step)
- When to file a patent application: priority date, Patent Cooperation Treaty (PCT)
- Where to protect an invention
- Different IPR strategies

The team and the early decisions

- The creation of the founders' team
- Types and characteristics of founders' teams
- Founders' decisions and their consequences
- Frequent mistakes and suggestions deriving from experience

The economic and financial aspects of a startup

- The fundamental economic and financial operations of a technology-based startup
- The structures of the financial statements
- Income Statement, Balance Sheet, Cash Flow
- Evaluation of the value of the company
- Sources and cost of capital

[back to **Summary**]

Funding a startup

- Different sources of funds: Angel Investors and Venture Capital
- Investment companies and funds: how they work
- How and what investors evaluate
- The investment agreements between investors and startups
- New ventures' funding options

References:

- Noam Wasserman (2013) The Founder's Dilemmas: Anticipating and Avoiding the Pitfalls That Can Sink a Startup, Princeton University Press.
- Thomas R. Ittelson (2009), Financial Statements: A Step-by-Step Guide to Understanding and Creating Financial Reports, Career Press.
- Hall, J., & Hofer, C. W. (1993). Venture capitalists' decision criteria in new venture evaluation. Journal of Business Venturing, 8(1), 25-42.

Schedule and room: please, see Class Schedule

Enrollment:

To attend the course registration is compulsory by using the Moodle platform of the PhD Course (in order to enter the Moodle platform click on "dettagli" of the course at the page http://www.cdii.dii.unipd.it/corsi). Once you are registered, if you cannot attend the course, please inform the lecturer.

Examination and grading: Attendance is required for at least 70% of the lecture hours (i.e. 14 hours). Final evaluation will be based on the discussion of a case study of a technology-based startup.

TSK 2. Python programming for Scientific Engineering

Course Area: Transversal Skills

Credits: 5

Instructor: Dr. Stefano Michieletto

e-mail: michieletto@dei.unipd.it

Aim: "Python is an easy to learn and powerful programming language." Python is becoming more and more popular for scientific applications such as machine learning, integrate and interpolate numerical information, manipulate and transform data. The first objective of the course is to become familiar with Python syntax, environments and basic libraries. The learner will be guided in performing basic inferential data analyses and introduced to the application of common machine learning algorithms.

Topics:

- A Quick Tour of Python Language Syntax
    - Python Basic Uses
    - What is different in Python?
- Modules and Packages
    - NumPy: Numerical Python
    - Pandas: Labeled Column-Oriented Data
    - Matplotlib: MATLAB-style scientific visualization
    - SciPy: Scientific Python

References:

[1] J. VanderPlas, "A Whirlwind Tour of Python", O'Reilly Media Inc. 2016. [Online: https://www.oreilly.com/programming/free/files/a-whirlwind-tour-of-python.pdf]

[2] J. VanderPlas, "Python Data Science Handbook: Essential Tools for Working with Data" O'Reilly Media Inc. 2017.

[3] B. Miles, "Begin to Code with Python", Pearson Education, Inc. 2018. [Online: https://aka.ms/BeginCodePython/downloads]

[4] Z. Shaw, "Learn Python the Hard Way", Addison-Wesley. 2014.

[5] L. Ramalho "Fluent Python", O'Reilly Media Inc. 2015.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Backgrounds in computing with some object-oriented programming language: C++, Java, MATLAB, etc. If you are starting from scratch, please have a look at [3] or [4].

[back to **Summary**]

Examination and grading: Homework assignments and final project.

IE 1. Statistical Methods

Course Area: Information Engineering

Credits: 6

Instructor: Dr. Lorenzo Finesso

e-mail: lorenzo.finesso@unipd.it

Aim: The course will present a small selection of statistical techniques which are widespread in applications. The unifying power of the information theoretic point of view will be stressed.

Topics:

- *Background material.* The noiseless source coding theorem will be quickly reviewed in order to introduce the basic notions of entropy and I-divergence. (a.k.a. relative entropy, Kullback-Leibler distance) between two probability measures.
- *Divergence minimization problems.* Three I-divergence minimization problems will be posed and, via examples, they will be connected with basic methods of statistical inference: ML (maximum likelihood), ME (maximum entropy), and EM (expectation-maximization).
- *Multivariate analysis methods.* The three standard multivariate methods, PCA (principal component analysis), Factor Analysis, and CCA (canonical correlations analysis) will be reviewed and their connection with divergence minimization discussed. Applications of PCA to least squares (PCR principal component regression, PLS Partial least squares). Approximate matrix factorization and PCA, with a brief detour on the approximate Nonnegative Matrix Factorization (NMF) problem. The necessary linear algebra will be reviewed.
- *EM methods.* The Expectation-Maximization method will be introduced as an algorithm for the computation of the Maximum Likelihood (ML) estimator with partial observations (incomplete data) and interpreted as an alternating divergence minimization algorithm à la Csiszár Tusnády.
- *Applications to stochastic processes.* Introduction to HMM (Hidden Markov Models). Maximum likelihood estimation for HMM via the EM method. If time allows: derivation of the Burg spectral estimation method as solution of a Maximum Entropy problem.

References:

A set of lecture notes and a complete list of references will be posted on the web site of the course.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: familiarity with basic linear algebra and probability.

[back to **Summary**]

Examination and grading: Homework assignments

[back to **Summary**]

IE 2. Statistics for Engineers

Course Area: Information Engineering

Credits: 5

Instructors: Prof. Luigi Salmaso, University of Padova, Prof. Arne Bathke, University of Salzburg.

e-mail: luigi.salmaso@unipd.it

Important note: registration mandatory; if you are interested to take this class (please see website https://phd.dei.unipd.it/courses/)

Aim: The course is an introduction to statistical methods most frequently used for experimentation in Engineering. Lectures are planned both in the classroom and in computer lab also for an introduction to the use of the following statistical software:

- R
- MINITAB (licensed to University of Padova)
- NPC TEST.

Topics:

1. Elements of univariate statistical methods:
    a. Elements of descriptive statistics: frequency, indices of synthesis (position, variability and shape) and graphical representations (histogram, boxplot, scatterplot).
    b. Elements of probability theory: discrete and continuous probability distributions.
    c. Elements of statistical inference: sampling distributions, point and interval estimation, hypothesis testing, One-way ANOVA, Multi-Way ANOVA, Factorial Designs.
2. Statistical Modelling:
    a. Experiments and observational studies, regression, residuals versus error terms, standard errors, generalized least squares, normal theory of regression, the F-test, path models, inferring causation from regression, multivariate regression and logit models, latent variables, nonparametric tests.

Bibliography

1. Stark, P.B., 1997. SticiGui: Statistics Tools for Internet and Classroom Instruction with a Graphical User Interface.
2. Montgomery DC, Design and Analysis of Experiments, 2010, Wiley.
3. Lattin J, Carroll JD, Green PE, Analyzing Multivariate Data, 2003,Duxbury Applied Series.
4. Johnson RA, Wichern DW, Applied Multivariate Statistical Analysis,1998, Prentice Hall; 4th edition.

[back to **Summary**]

5. Hollander and Wolfe, Nonparametric Statistical Methods, 2nd edition, 1999, Wiley Series in Probability and Statistics.
6. Shumway RH, Stoffer DS, Time Series Analysis and Its Applications (With R Examples), 2nd Edition, 1998, Springer Texts in Statistics, NewYork.
7. Adhoc material by Lecturer.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Examination and grading: Attendance is required for at least 2/3 of the lecture hours. Final evaluation will be based on the discussion of a case study within the individual PhD project.

IE 3. Computational Inverse Problems

Course Area: Information Engineering

Credits: 5

Instructor: Prof. Fabio Marcuzzi, Dept. of Mathematics, University of Padova.

e-mail: marcuzzi@math.unipd.it

Aim: We study numerical methods that are of fundamental importance in computational inverse problems. Real application examples will be given for distributed parameter systems in continuum mechanics. Computer implementation performance issues will be considered as well.

Topics:

- definition of inverse problems, basic examples and numerical difficulties.
- numerical methods for QR and SVD and their application to the square-root implementation in PCA, least-squares, model reduction and Kalman filtering; recursive least-squares;
- regularization methods;
- numerical algorithms for nonlinear parameter estimation: Gauss-Newton, Levenberg-Marquardt,
- examples with distributed parameter systems in continuum mechanics; HPC implementations

References:

[1]F.Marcuzzi "Analisi dei dati mediante modelli matematici", http://www.math.unipd.it/~marcuzzi/MNAD.html

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements:

- basic notions of linear algebra and, possibly, numerical linear algebra.
- the examples and homework will be in Python (the transition from Matlab to Python is effortless).

Examination and grading: Homework assignments and final test.

IE 4. Heuristics for Mathematical Optimization

Course Area: Information Engineering

Credits: 5

Instructor: Prof. Domenico Salvagnin

e-mail: dominiqs@gmail.com, domenico.salvagnin@unipd.it

Aim: Make the students familiar with the most common mathematical heuristic approaches to solve mathematical/combinatorial optimization problems. This includes general strategies like local search, genetic algorithms and heuristics based on mathematical models.

Topics:

- Mathematical optimization problems (intro).
- Heuristics vs exact methods for optimization (intro).
- General principle of heuristic design (diversification, intensification, randomization).
- Local search-based approaches.
- Genetic/population based approaches.
- The subMIP paradigm.
- Applications to selected combinatorial optimization problems: TSP, QAP, facility location, scheduling.

References:

[1] Gendreau, Potvin "Handbook of Metaheuristics", 2010

[2] Marti, Pardalos, Resende "Handbook of Heuristics", 2018

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements:

- Moderate programming skills (on a language of choice)
- Basics in linear/integer programming.

Examination and grading: Final programming project.

[back to **Summary**]

IE 5. Statistical Learning for Big Data in Medicine

Course Area: Information Engineering

Credits: 4

Instructor: Prof. Andrea Facchinetti (Department of Information Engineering, University of Padova), Dr. Martina Vettoretti (Department of Information Engineering, University of Padova)

e-mail: facchine@dei.unipd.it

Aim: The course is intended to provide a better understanding of the methodologies used in the analysis of big data in medical applications and epidemiology.

Topics:

- Types of clinical studies (randomized clinical trials, retrospective studies, longitudinal studies), definition of exposures and main outcomes (incidence, prevalence, risk ratio, odds ratio);
- Logistic regression to link covariates to the main outcome: definition and properties, parameter estimation via maximum likelihood, coefficient interpretation, goodness of fit tests, covariate selection;
- Survival analysis: definition of lifetime, survival, and hazard functions; univariate nonparametric and parametric survival analysis; multivariate survival analysis with parametric and semiparametric (Cox) proportional hazard models;
- Hands-on: analysis of big data collected in medical research and implementation of logistic / survival models for the prediction of clinical outcomes

References:

- A set of lecture notes and a complete list of references will be made available by the Lecturer
- The Elements of Statistical Learning: Data Mining, Inference, and Prediction (Second Edition, 2009) by Trevor Hastie, Robert Tibshirani, Jerome Friedman

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basics of probability theory and statistics; basics of Matlab programming.

Examination and grading: Final project consisting in the development of logistic / survival models on a given dataset.

IE 6. Model Predictive Control with Case Studies in Automotive and Biomedicine

Course Area: Information Engineering

Credits: 5

Instructors: Prof. Ruggero Carli, Dr. Mattia Bruschetta, Dr. Simone Del Favero, Department of Information Engineering, University of Padova

e-mail : carlirug@dei.unipd.it, mattia.bruschetta@dei.unipd.it, simone.delfavero@unipd.it

Aim: The course will provide the basic knowledge of Model Predictive Control (MPC). The course will also present some practical examples related to Automotive and Bioengineering applications.

Topics:

1. Introduction to MPC: main ideas behind MPC control.
2. State Space Models. Prediction and Current State Estimation (Kalman Filter);
3. Linear Quadratic Problem, the Infinite Horizon LQ Problem, Convergence of the Linear Quadratic Regulator.
4. Model Predictive Control - Regulation: Formulation, Dynamic Programming Solution, Stability properties, MPC for Unconstrained Systems, MPC for Systems with Control Constraints, MPC for Systems with Control and State Constraints, Suboptimal MPC, Tracking.
5. Robust MPC and explicit MPC: Types of Uncertainty, Nominal robust-ness, tube-based robust MPC, Explicit Control Laws for Constrained Lin-ear Systems.
6. Fast Nonlinear MPC: Direct and Indirect methods for NLP, Condensing, Sequential Quadratic Programming, Real Time Iteration Scheme.
7. Automotive case studies: Motion Cueing Algorithms, Virtual Rider, Autonomous Driver.
8. Biomedicine case study: the Artificial Pancreas. The Blood Glucose Regulation Problem, possible MPC Approaches (modular MPC, zone MPC, non-linear MPC), Clinical Testing.

References:

[1] J. B. Rawlings and D. Q. Mayne. Model predictive control : Theory and design. Nob Hill Publisher.

Other material and research papers will be available online for download.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basic Calculus and Linear Algebra.

Examination and grading: Homework and take home exam

[back to **Summary**]

IE 7. Fluid mechanics for the functional assessment of cardiovascular devices

Course Area: Information Engineering

Credits: 4

Instructor: Prof. Francesca Maria Susin, Dept. of Civil, Environmental and Architectural Engineering (DICEA)

e-mail: francescamaria.susin@unipd.it

Aim:

The course is intended to give a survey of research approaches for the assessment of cardiovascular medical devices. Emphasis will be given to methods and techniques adopted for in vitro analysis of hemodynamic performance of prosthetic heart valves and total artificial heart.

Topics:

Review of basic fluid mechanics concepts. Fluid mechanics of prosthetic heart valves (PHVs) and ventricular assist devices (VADs). Pulse duplicators for in vitro testing of PHVs and mock circulation loops for pre-clinical evaluation of VADs. Experimental techniques for the assessment of PHVs and VADs performance. CFD for functional assessment of PHVs and VADs.

References:

[1] M. Grigioni, C. Daniele, G. D'Avenio, U. Morbiducci, C. Del Gaudio, M. Abbate and D. Di Meo. Innovative technologies for the assessment of cardiovascular medical devices: state of the art techniques for artificial heart valve testing. Expert Rev. Medical Devices, 1(1) : 81-93, 2004.

[2] K.B. Chandran, A.P. Yoganathan and S.E. Rittgers. Bio fluid Mechanics: the human circulation. CRC Press, Boca Raton, FL, 2007.

[3] A.P. Yoganathan, K.B. Chandran and F. Sotiropoulos. Flow in prosthetic heart valves: state of the heart and future directions. Annals of Biomedical Engineering, 33(12) : 1689-1694, 2005.

[4] A.P. Yoganathan, Z. He and S. Casey Jones. Fluid mechanics of heart valves.

[5] A.P. Yoganathan and F. Sotiropoulos. Using computational fluid dynamics to examine the hemodynamics of artificial heart valves. Business briefing: US cardiology 2004 : 1-5, 2004.

[6] V. Barbaro, C. Daniele and M. Grigioni. Descrizione di un sistema a flusso pulsatile per la valutazione delle protesi valvolari cardiache. ISTI-SAN Report 91/7, Rome, Italy, 1991 (in Italian).

[7] M. Grigioni, C. Daniele, C. Romanelli and V. Barbaro. Banco di prova per la caratterizzazione di dispositivi di assistenza meccanica al circolo. ISTISAN Report 03/21, Rome, Italy, 2003 (in Italian).

[back to **Summary**]

[8] M.J. Slepian, Y. Alemu, J.S. Soares. R.G. Smith, S. Einav and D. Bluestein. The Syncardia total arti cial heart: in vivo, in vitro, and computational modeling studies. Journal of Biomechanics,46 (2013): 266-27, 2013.

Further references will be given during the course.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Fundamentals of Fluid Dynamics.

Examination and grading: Homework assignment with final discussion.

IE 8. Diagnostics of Electron Devices

Course Area: Information Engineering

Credits: 5

Instructor: Prof. Giovanna Mura, Prof. Massimo Vanzi - Department of Electrical and Electronic Engineering (DIEE), University of Cagliari.

e-mail: gmura@diee.unica.it, vanzi@diee.unica.it

Aim: this course provides an overview of the Failure Analysis techniques for the diagnostics of electron devices. Failure analysis is the process of analyzing the failed electron devices to determine the reason for degraded performance or catastrophic failure and to provide corrective actions able to x the problem. It is a proactive tool with three fundamental tasks: 1) Technical/scientific: 2) Technological 3) Economical. The purpose of this course is to teach what Failure Analysis should be and should do, to show how and why it often does not, to state that F.A. has Logics and has Rules.

Microscopy, in its several forms (optical, electron, scanning, transmission, emission, ionic) and tools is the playground for practical FA, and its fundamentals will be described. Device basic technology, working principle and failure physics are the other pillars for a successful study.

Several case studies will be proposed with the aim to demonstrate that if sometimes Failure Analysis looks unclear or not problem solving is merely because it was badly conducted.

Topics:

1. Reverse engineering
2. Failure modes and failure mechanisms
3. Principles and fundamental methods in Electron Microscopy
4. Methodology for the Failure Analysis

References: Failure Analysis of Integrated Circuits - Tools and Techniques, Springer International Series - Lawrence C. Wagner.

Slides

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Electron Devices, Microelectronics, Optoelectronics devices.

Examination and grading: Written test/ presentation of a report at the end of the course

[back to **Summary**]

IE 9. Physics and Operation of Heterostructure-Based Electronic and Optoelectronic Devices

Course Area: Information Engineering

Credits: 5

Instructors: Prof Enrico Zanoni, Prof. Matteo Meneghini, Dr. Carlo De Santi, DEI, University of Padova.

e-mail: zanoni@dei.unipd.it, menego@dei.unipd.it, desantic@dei.unipd.it

Aim: This course provides an introduction to the physics and operating principles of advanced electronic and optoelectronic devices based on compound semiconductors. These devices are particularly important for several applications: high electron mobility transistors (HEMTs) represent excellent devices for the realization of high frequency communication systems, radars, satellite applications, and high efficiency power converters. On the other hand, LEDs and lasers are high-efficiency monochromatic light sources, that can be used both for lighting applications (with a considerable energy saving), in the biomedical field, and in in photochemistry. Special focus will be given to Gallium Nitride (GaN) based devices, that represent the most promising devices for future power electronics applications. This course will focus on the main aspects related to the physics of heterostructures, on the recombination processes in semiconductors, on carrier transport in heterostructures, on the structure and operating principles of MESFET, HEMTs, GITs, on the trapping and reliability in compound semiconductor devices, on the operating principles of LEDs and lasers, and on parasitics and reliability in LEDs and lasers. An overview of real applications highlighting the capabilities of these devices will also be given.

Topics:

- physics of heterostructures, band diagrams, carrier transport in heterostructures;
- recombination processes in semiconductors; properties of compound semiconductors;
- basic structure of heterojunction transistors, MESFET, HEMT, GIT; parasitics and reliability in HEMTs, LEDs and lasers;
- operating principles of LEDs and lasers;
- methods for advanced characterization of heterojunction based devices; applications of GaN based HEMTs, LEDs and lasers;
- modeling of semiconductor-based devices

References:

Umesh Mishra, Jasprit Singh, Semiconductor Device Physics and Design, Springer, 2008

Ruediguer Quay, Gallium Nitride Electronics, Springer 2008.

Tae-Yeon Seong, Jung Han, Hiroshi Amano, Hadis Morko, III-Nitride Based Light Emitting Diodes and Applications, Springer 2013

[back to **Summary**]

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Introductory course of device physics: Microelectronics, Optoelectronic and Photovoltaic Devices.

Examination and grading: Written test at the end of the course.

IE 10. Embedded Design with FPGA

Course Area: Information Engineering

Credits: 5

Instructors: Dr. Andrea Stanco and Prof. Daniele Vogrig, Department of Information Engineering, University of Padova

e-mail: andrea.stanco@dei.unipd.it, vogrig@dei.unipd.it

Aim: The course aims at teaching how to practically use System-on-a-Chip (FPGA+CPU) as a potentially application to academic research topics. 75% of the course will be held in a dedicated laboratory to deal with the programming of a Pynq-Z1 board.

Topics:

- Recap on basic of Digital Design. Digital Design Flow (HDL language and HLS). Introduction to VHDL program language.
- Introduction to FPGA and Zynq SoC.
- Introduction to Vivado System Design environment. Time domains, time violations, metastability, system constraints.
- Introduction to SDK environment
- Information exchange between processor and programmable logic. Hardware and Software interrupts.
- Communication between SoC and the outside world.
- Case studies

References:

[1] Xilinx, Vivado Design Suite User Guide, UG893 (v2019.1), https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_1/ug893-vivado-ide.pdf

[2] Xilinx, Xilinx Software Development Kit (SDK) User Guide, https://www.xilinx.com/support/documentation/sw_manuals/xilinx2015_1/SDK_Doc/index.html

Other material will be pointed out in class and available online for download

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basic knowledge of digital electronics. Knowledge of program language (e.g. C/C++). No VHDL knowledge or experience on FPGAs is required.

[back to **Summary**]

Examination and grading: Homework assignments and final project.

IE 11. Introduction to Information Theory

Course Area: Information Engineering

Credits: 4

Instructor: Prof. Deniz Gunduz

e-mail: d.gunduz@imperial.ac.uk

Important note: the first half of the course will be held by Prof. Gunduz in two, four-hour lectures at the University of Padova (see schedule below). The remaining lectures will be offered by Prof. Gunduz remotely using the videoconferencing room. Date and time of these remaining lectures will be agreed between the instructor and attendees during the first two lectures.

Aim: The aim of this course is to introduce basic information theoretic concepts to students. We will start by introducing entropy, divergence, and, mutual information, and their mathematical properties. The rest of the course will be dedicated to illustrating engineering applications of these seemingly abstract quantities. We will see that entropy corresponds to the ultimate limit in data compression, divergence provides the best error exponent in hypothesis testing (i.e., binary classification), and mutual information sets the limit of how much data one can transmit reliably over a noisy communication channel.

Syllabus

- **Information measures**
    - Entropy, divergence, mutual information
    - Properties of information measures (chain rule, data processing inequality, convexity)
- **Lossless data compression**
    - Asymptotic equipartition property (AEP)
    - Kraft inequality, Huffman coding and its optimality
- **Information theory and learning**
    - Method of types, universal source coding, large deviations: Sanov's theorem
    - Hypothesis testing, Stein's lemma, Chernoff exponent
- **Channel coding**
    - Channel capacity theorem, achievability, joint AEP
    - Converse to channel coding theorem, feedback capacity, Joint source-channel coding

References:

[1] R. B. Ash, *Information Theory*, Dover, 1990.

[2] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley, 1991.

[3] R. G. Gallager, *Information Theory and Reliable Communication*, Wiley, 1968.

[back to **Summary**]

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basic knowledge of probability theory will be assumed.

Examination and grading: Grades will be based on a final exam.

IE 12. Machine Learning for Wireless Communication Systems

Course Area: Information Engineering

Credits: 4

Instructor: Dr. Paolo Dini, Senior Researcher, Centre Tecnologic de Telecomunicacions de Catalunya

e-mail: paolo.dini@cttc.es

Aim: The course will introduce the main paradigms used in the design of next-generation mobile systems, such as the network densification and softwarization, to provide broadband communications and support vertical markets. The issues on controlling such an elastic mobile network architecture will be discussed. We will, then, identify suitable machine learning tools enabling automatic network monitoring and management. For this, fundamentals of Reinforcement Learning and Artificial Neural Networks will be given together with their practical application to the identified mobile networking problems.

Topics:

- Introduction of next-generation mobile network scenarios and architectures
  - network densification paradigm
  - network softwarization paradigm
  - veticals
- Identification of machine learning tools for networking
- Fundamentals of Reinforcement Learning
  - approximated Dynamic Programming
  - Temporal-Difference methods
  - Deep-Q networks
- Fundamentals of Artificial Neural Network architectures
  - Multi-layer perceptrons
  - Recurrent neural networks
  - Convolutional neural networks
  - Auto-encoders
- Mobile network on-line optimization methods
- Mobile traffic characterization and modeling

References:

[1] Anwer Al-Dulaimi, Xianbin Wang, Chih-Lin I, 5G Networks: Fundamental Requirements, Enabling Technologies, and Operations Management, October 2018, Wiley-IEEE Press

[2] R. Boutaba, M.A. Salahuddin, N. Limam, et al., A comprehensive survey on machine learning for networking: evolution, applications and research opportunities, Elsevier J. Internet Serv Appl (2018) 9: 16

[back to **Summary**]

[3] Richard Sutton, Andrew G. Barto, Reinforcement Learning: An Introduction, MIT press, 2017

[4] Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006

[5] Ian Goodfellow, Yoshua Bengio, Aaron Courville, Deep Learning, MIT press

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basic knowledge of probability theory, random processes, python scripting.

Examination and grading: Each student will develop a final project, possibly related to his/her research activity, addressing some topic presented in the Course.

IE 13. Introduction to Reinforcement Learning

Course Area: Information Engineering

Credits: 4

Instructor: Dr. Juan José Alcaraz Espín, Associate Professor, Technical University of Cartagena, Spain.

e-mail: juan.alcaraz@upct.es

Aim: The course will provide an introduction to the field of reinforcement learning, covering its mathematical foundations and the description of the most relevant algorithms. The main concepts and techniques will be illustrated with Python code and application examples in telecommunications and other related areas. The students will acquire hands-on experience with the proposed assignments in which they will have to implement Python code for solving several challenges and exercises. The course will start with the basic concepts of learning in sequential decision problems, formalized in the multi-armed bandit (MAB) problem and its variants. Then, the Markov decision processes (MDPs), which generalize the MAB problem, will be introduced. The objective of reinforcement learning (RL) is to find approximate solutions to MDPs. The main RL approaches will be presented incrementally: 1) tabular methods, which are capable of addressing relatively small problems, 2) value function approximation, which allows scaling up previous algorithms to larger problems, and 3) policy gradient algorithms which follow a different scaling approach and can be used in combination with value function approximation (Actor-Critic methods).

Topics:

Unit 1. Introduction to Reinforcement Learning

Unit 2. Multi-Armed Bandits: Stochastic Bandits, Boltzmann Exploration, UCB algorithms, Thompsom Sampling, Contextual Bandits.

Unit 3. Markov Decision Processes: Stochastic Shortest Path problems. Policy Iteration. Value Iteration. MDPs with discount.

Unit 4. Tabular Methods: Monte Carlo Method, Temporal Difference, Off-policy algorithms, Planning at decision time.

Unit 5. Value Function Approximation (VFA) Methods: Linear VFA, Monte Carlo with VFA, TD methods with VFA.

Unit 6. Policy Gradient Algorithms: Score functions, Policy Gradient Theorem, Monte Carlo Policy Gradient, Actor-Critic Policy Gradient.

Unit 7 (Optional) Evolutionary Algorithms

References:

[1] Reinforcement Learning: An Introduction, Second Edition, Richard S. Sutton and Andrew G. Barto, MIT Press, Cambridge, MA, 2018.

[back to **Summary**]

[2] Approximate Dynamic Programming: Solving the Curses of Dimensionality, Second Edition, Warren B. Powell, Wiley, 2011.

[3] Dynamic Programming and Optimal Control Vol I and Vol II, 4th Edition, Dimitri P. Bertsekas, Athena Scientific, 2012.

[4] Algorithms for Reinforcement Learning, Csaba Szepesvári, Morgan and Claypool, 2010.

[5] Reinforcement Learning and Optimal Control, Dimitri P. Bertsekas, Athenea Scientific, 2019.

[6] Markov Decision Processes: Discrete Stochastic Dynamic Programming, Martin L. Puterman, Wiley, 2006.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course Requirements: Basics of linear algebra, probability theory, Python scripting

Examination and Grading: The grading will be based on the students' solutions to the proposed assignments.

IE 14. Information Theoretic Models in Security

Course Area: Information Engineering

Credits: 5

Instructor: Prof. Nicola Laurenti, Department of Information Engineering

e-mail: nil@dei.unipd.it

Aim: The class aims at providing the students with an information theoretic framework that will allow formal modeling, understanding of the fundamental performance limits, and derivation of unconditionally secure mechanisms for several security-related problems.

Topics:

- *All or nothing: security without compromise.* Quantitative definition of security. Unconditional security. Distinguishability. Universally composable security
- *The Holy Grail of perfect secrecy.* Shannon's cipher system. Perfect secrecy. Ideal secrecy. Practical secrecy. The guessing attack.
- *Secrecy without cryptography.* The wiretap channel model. Rate-equivocation pairs. Secrecy rates. Secrecy capacity for binary, Gaussian and fading channel models.
- *Security from uncertainty.* Secret key agreement from common randomness on noisy channels. Information theoretic models and performance limits of quantum cryptography.
- *Who's who?* An information theoretic model for authentication in noisy channels. Signatures and fingerprinting.
- *The gossip game.* Broadcast and secrecy models in multiple access channels. The role of trusted and untrusted relays.
- *Secrets in a crowd.* Information theoretic secrecy in a random network with random eavesdroppers. Secrecy graphs and large networks secrecy rates.
- *A cipher for free?* Information theoretic security of random network coding.
- *The jamming game.* Optimal strategies for transmitters, receivers and jammers in Gaussian, fading and MIMO channels.
- *Alea iacta est.* Secure and true random number generation. Randomness extractors and smooth guessing entropy
- *Writing in sympathetic ink*. Information theoretic models of steganography, watermarking and other information hiding techniques.
- *Leaky buckets and pipes.* Information leaking and covert channels. Timing channels.
- *The dining cryptographers.* Privacy and anonymity. Secure multiparty computation.
- *Information theoretic democracy.* Privacy, reliability and verifiability in electronic voting systems.
- *The Big Brother.* An information theoretic formulation of database security: the privacy vs utility tradeoff. Differential privacy.

[back to **Summary**]

References:

- Y. Liang, H.V. Poor, and S. Shamai (Shitz), *Information Theoretic Security*, Now, 2007.
- M. Bloch, J. Barros, *Physical-Layer Security: from Information Theory to Security Engineering*, Cambridge University Press, 2011.

A short list of reference papers for each lecture will be provided during class meetings.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basic notions of Information Theory (e.g., those from the *Telecomunicazioni* class in the *Corso di Laurea in Ingegneria dell'Informazione*)

Examination and grading: Each student (or small group of students) must submit a project, and grading will be based on its evaluation. Students are encouraged to work from an information theoretic point of view on a security problem related to their research activities.

IE 15. Modeling and Simulation of Complex & Multi-Disciplinary Dynamical Systems

Course Area: Information Engineering

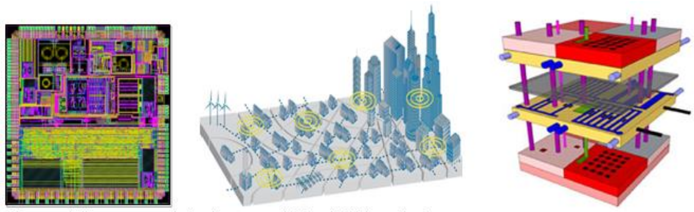
Credits: 4

Instructor: Luca Daniel, Professor of Electrical Engineering and Computer Science, Massachusetts Institute of Technology

e-mail: luca@mit.edu

URL: projects abstracts and video presentations with demos by students that took this class at MIT

Aim: Many complex systems found in nature/society and studied by social/economical/physical scientists (e.g. the human cardiovascular system, the brain neural network, biological systems, the geophysical network of oil/water/gas reservoirs, social networks), or developed by engineers (e.g. labs on chips, iPads, magnetic resonance scanners, nationwide electrical/gas/oil transportation network, supply chains, currency/stock markets, buildings/automotive/aircraft frames) can be viewed as large collections of interconnected dynamical system components. The performance of such complex systems typically depends on complicated constitutive and conservation relations between components, as well as on random uncertainties.



Power delivery network for Integrated Circuit (IC) or city/state   Heat sink for 3D IC

Blood/Oil delivery network   Building/Mechanical frames

The **goal** of this course is to provide students with a working hands-on knowledge of the state of the art in modeling, simulation, model order reduction and uncertainty quantification techniques. Examples will be drawn from a large variety of complex and multi-disciplinary dynamical systems, as well as from student proposed applications, helping them with their own research projects in different engineering and science disciplines that deal with complex systems.

Topics:

Upon completion of this course students will be able to:

- Recognize and formulate mathematical frameworks (e.g. conservation laws and constitutive equations) common to a lot of multi-disciplinary complex dynamical systems.
- Select, modify and implement an appropriate steady state solver (e.g. sparse LU vs. iterative methods) for their linear or linearized complex system description.

[back to **Summary**]

- Select, implement and modify an appropriate strategy to facilitate initialization and convergence of a Newton solver for their nonlinear complex system.
- Select and implement an appropriate technique (e.g. explicit fix time step vs. implicit adaptive time steps) for the time domain simulation of their complex system.
- Select and implement an appropriate strategy (e.g. Shooting Newton or Harmonic Balance) for period state analysis of their complex system (e.g. vibrations in mechanical/structural frames, radio frequency circuits, heart beat cycles).
- Select and implement an appropriate strategy to "reduce" automatically models of system components generated for instance by large PDE solvers' discretizations, while preserving input/output accuracy for a range of parameter values, as well as important physical properties. Select and implement an appropriate strategy to "generate" automatically stable parameterized reduced models of system components from input/output measurements.
- Use parameterized reduced order models of system components in order to accelerate optimization, inverse problems in complex systems.
- Select and implement uncertainty quantification techniques for stochastic simulation of complex systems affected by random variations in geometries and material properties.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Calculus, differential equations, linear algebra as well as some basic programming experience in Matlab, or other programming languages for scientific computing.

Examination and grading: Students will be working in small teams on a course-long project involving modeling and simulation of a complex system either self-proposed from their own field of research, or chosen from a few examples developed in class. Time in class will include short lectures interleaved by numerous interactive and hands-on activities coordinated by the instructor and supporting the self-proposed projects. Final evaluations will be based on in-class work and interaction with the staff during the course as well as on a final live project demo presentation and report. The focus of the course will not be on mathematical formalism and rigorous theorem proving, but rather on developing general intuition, creativity, practical implementation and model debugging skills.

[back to **Summary**]

IE 16. Elements of Deep Learning

Course Area: Information Engineering

Credits: 6

Instructor: Dr. Gian Antonio Susto

e-mail: gianantonio.susto@dei.unipd.it,

Aim: The course will serve as an introduction to Deep Learning (DL) for students who already have a basic knowledge of Machine Learning. The course will move from the fundamental architectures (e.g. CNN and RNN) to hot topics in Deep Learning research.

Topics:

- Introduction to Deep Learning: context, historical perspective, differences with respect to classic Machine Learning.
- Feedforward Neural Networks (stochastic gradient descent and optimization).
- Convolutional Neural Networks.
- Neural Networks for Sequence Learning.
- Elements of Deep Natural Language Processing.
- Elements of Deep Reinforcement Learning.
- Unsupervised Learning: Generative Adversarial Neural Networks and Autoencoders.
- Laboratory sessions in Colab.
- Hot topics in current research.

References:

[1] Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein GAN. CoRR, abs/1701.07875.

[2] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. CoRR, abs/1409.0473.

[3] I. Goodfellow, Y. Bengio, A. Courville ‘Deep Learning’, MIT Press, 2016

[4] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A.C., & Bengio, Y. (2014). Generative Adversarial Nets. NIPS.

[5] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. Neural computation, 9 8, 1735-80.

[6] Kalchbrenner, N., Grefenstette, E., & Blunsom, P. (2014). A Convolutional Neural Network for Modelling Sentences. ACL.

[7] Krizhevsky, A., Sutskever, I., & Hinton, G.E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Commun. ACM, 60, 84-90.

[8] LeCun, Y. (1998). Gradient-based Learning Applied to Document Recognition.

[back to **Summary**]

[9] Mikolov, T., Sutskever, I., & Chen, K. (2013). Representations of Words and Phrases and their Compositionality.

[10] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P. (2010). Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. Journal of Machine Learning Research, 11, 3371-3408.

[11] Zaremba, W., Sutskever, I., & Vinyals, O. (2014). Recurrent Neural Network Regularization. CoRR, abs/1409.2329.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basics of Machine Learning and Python Programming.

Examination and grading: Final project.

IE 17. Applied Functional Analysis and Machine Learning

Course Area: Information Engineering

Credits: 7

Instructor: prof. Gianluigi Pillonetto

e-mail: giapi@dei.unipd.it

Aim: The course is intended to give a survey of the basic aspects of functional analysis, machine learning, regularization theory and inverse problems.

Topics: Review of some notions on metric spaces and Lebesgue integration}: Metric spaces. Open sets, closed sets, neighborhoods. Convergence, Cauchy sequences, completeness. Completion of metric spaces. Review of the Lebesgue integration theory. Lebesgue spaces.

Banach and Hilbert spaces: Finite dimensional normed spaces and subspaces. Compactness and finite dimension. Bounded linear operators. Linear functionals. The finite dimensional case. Normed spaces of operators and the dual space. Weak topologies. Inner product spaces and Hilbert spaces. Orthogonal complements and direct sums. Orthonormal sets and sequences. Representation of functionals on Hilbert spaces.

Compact linear operators on normed spaces and their spectrum: Spectral properties of bounded linear operators. Compact linear operators on normed spaces. Spectral properties of compact linear operators. Spectral properties of bounded self-adjoint operators, positive operators, opera- tors defined by a kernel. Mercer Kernels and Mercer theorem.

Reproducing kernel Hilbert spaces, inverse problems and regularization theory: Representer theorem. Reproducing Kernel Hilbert Spaces (RKHS): definition and basic properties. Examples of RKHS. Function estimation problems in RKHS. Tikhonov regularization. Primal and dual formulation of loss functions. Regularization networks. Consistency/generalization and relationship with Vapnik's theory and the concept of V-gamma dimension. Support vector regression and classification.

References:

[1] W. Rudin. Real and Complex Analysis, McGraw Hill, 2006

[2] C.E. Rasmussen and C.K.I. Williams. Gaussian Processes for Machine Learning. The MIT Press, 2006

[3] H. Brezis, Functional analysis, Sobolev spaces and partial differential equations, Springer 2010

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

[back to **Summary**]

Course requirements: The classical theory of functions of real variable: limits and continuity, differentiation and Riemann integration, infinite series and uniform convergence. The arithmetic of complex numbers and the basic properties of the complex exponential function. Some elementary set theory. A bit of linear algebra.

Examination and grading: Homework assignments and final test.

IE 18. Applied Linear Algebra

Course Area: Information Engineering

Credits: 5

Instructors: Prof. Luca Schenato, Dipartimento di Ingegneria dell'informazione, Università di Padova (http://automatica.dei.unipd.it/people/schenato.html)

e-mail: schenato@dei.unipd.it

Aim: We study concepts and techniques of linear algebra that are important for applications with special emphasis on the topics: solution of systems of linear equations with particular attention to the analysis of the backward error and computational cost of the basic algorithms and matrix equation. A wide range of exercises and problems will be an essential part of the course and constitute homework required to the student.

Topics:

1. Vectors: inner products, norms, main operations (average, standard deviation, ...)
2. Matrices: matrix-vector and matrix-matrix multiplication, Frobenius norm,
3. Complexity, sparsity
4. Special matrices: Diagonal, Upper Triangular, Lower triangular, Permutation (general pair), inverse and orthogonal
5. A square and invertible: LU decomposition (aka gaussian elimination), LU-P decomposition, Cholesky decomposition
6. Ax=b via LU-P decomposition: forward and backward substitution
7. (sub)Vector spaces: definitions, span, bases (standard, orthogonal, orthonormal), dimension, direct sum, orthogonal complement, null space, orthogonal complement theorem
8. Gram-Smith orthogonalization and QR decomposition (square and invertible A, general non-square)
9. Ax=b via QR decomposition. LU-P vs QR
10. Linear maps: image space, kernel, column and row rank
11. Fundamental Theorem of Linear Algebra (Part I): rank-nullity Theorem, the 4 fundamental subspace
12. Eigenvalues/eigenvector and Shur decomposition
13. Projection matrices: oblique and orthogonal, properties
14. Positive semidefinite matrices: properties and quadratic functions square root matrix
15. Properties of A'A and AA' and Polar decomposition
16. Singular Value Decomposition: proofs and properties
17. Pseudo-inverse: definition and relation to SVD
18. Fundamental Theorem of Linear Algebra (Part II): special orthogonal basis for diagonalization
19. Least-Squares: definition, solution and algorithms

[back to **Summary**]

20. Ill-conditioned problems vs stability of algorithms, numerical conditioning of algorithms, numerical conditionings

Objectives:

- Theory: formal proofs of many results (theorem-proof type problems)
- Algorithms: understanding of most commonly used algortithm used in MATLAB and Python for Linear Algebra
- Implementation: MATLAB implementation of algorithms and performance evaluation on Big Data

References:

Textbooks and Internet Notes:

1. S. Boyd, L. Vanderberghe, "**Introduction to Applied Linear Algebra**", Cambridge University Press, 2018
2. G. Strang, "**The Fundamental Theorem of Linear Algebra**", *The American Mathematical Monthly*, vol. 100(9), pp. 848-855, 1993
3. G. Strang, "**Linear Algebra and Learning From Data",** Wellesley - Cambridge Press, 2019

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: A good working knowledge of basic notions of linear algebra as for example in [1]. Some proficiency in MATLAB.

Examination and grading: Grading is based on Homeworks, Written final exam, Short presentation based on a recent paper of Linear Algebra Algorithms for Big Data.

IE 19. Control of Multivariable Systems: A Geometric Approach

Course Area: Information Engineering

Credits: 5

Instructor: Prof. Andrea Serrani, The Ohio State University

e-mail: serrani.1@osu.edu

Aim: The goal of the course is to introduce the geometric theory of linear multivariable systems as a fundamental tool for the solution of relevant control problems, including disturbance rejection, non-interaction, fault detection and isolation, and tracking and regulation. Attention will be devoted to routines available in MATLAB for numerical implementation of the control algorithms presented in class. Design examples on a realistic model of an aerospace system will be introduced.

Topics:

1. **Background:** Subspaces, maps, factor spaces, projections.
2. **Systems Theory:** Controllability, observability, compensator design.
3. **Disturbance Decoupling:** Controlled invariance, controllability subspaces. Duality: Conditioned invariance, unknown-input observers.
4. **Eigenvalue Assignment under Invariance Constraints:** Multivariable zeros. Zero dynamics.
5. **Non-interacting Control:** Synthesis via dynamic extension. Duality: Fault detection and isolation.
6. **Tracking and Regulation:** Right-inversion. The regulator problem.

References: W.M. Wonham, "Linear Multivariable Control: A Geometric Approach," Springer-Verlag; Supplementary notes.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: A basic course in linear system theory and proficiency in linear algebra are required. Working knowledge in MATLAB/SIMULINK is needed for the design examples.

Examination and grading: Homework assignments and/or take-home final examination.

[back to **Summary**]

IE 20. Causal Inference for Complex Networks

Course Area: Information Engineering

Credits: 4

Instructor: Prof. Reza Arghandeh, Department of Electrical and Computer Engineering, Florida State University

e-mail: r.arghandeh@fsu.edu

Aim:

One of the notable analytical challenges of our century is the intricate complexity of systems that shape our civilization ranging from electricity networks to computer networks to biological networks and social networks due to all interdependency and interconnectivity among them. It is near impossible to understand complex network systems behavior unless we go beyond the classic machine learning and network science and develop a casual insight into the machinery behind different networks. Nevertheless, the notable differences in forms, scopes, components, and nature of different networks, most networks follow common cause and effect principles. This course provides a selection of concepts from information theory and causality inference domains to analyze complex networks considering their inherent interdependencies. During the course, students will be familiar with use cases from electric grids, roadways, and social networks.

Topics:

1. Motivating problems in complex networked systems. i) some analytical problems in smart grids. ii) some analytical problems in smart cities.
2. Elements of Graph Theory: i) overview of graphs. ii) path, connectivity, and weighted graphs. iii) metrics for graphs.
3. Causality Inference: i) causality language ii) theory of causation and intervention iii) state- of-the-art causality inference methods
4. Causality for Complex networks i) causality methods for large scale networks ii) example applications in smart grids

References:

[1] J. Pearl, Causality, Cambridge University Press, 2009.

[2] A. Barabasi, Network Science, Cambridge University Press , 2016.

[3] F. Bullo, Lectures on Network Systems, CreateSpase, 2018.Class lectures and other material and research papers will be available online for download.

Time table: Course of 16 hours.

[back to **Summary**]

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: familiarity with basic probability. Knowledge of network theory also helps, but it is not a requirement.

Examination and grading: a final project or a take-home exam.

IE 21. Real-Time Systems and Applications

Course Area: Information Engineering

Credits: 5

Instructor: Dr. Gabriele Manduchi, Consiglio Nazionale delle Ricerche

e-mail: gabriele.manduchi@igi.cnr.it

Aim: The course will provide an insight in the realm of real-time system. Knowledge in this field is normally fragmented and scattered among different engineering disciplines and computing sciences, and the aim of the course is present aspects related to theory and practice in a way which is holistic enough to prepare graduates to embark on the development of real-time systems, frequently complex and imposing safety requirements

Topics:

- Introduction via a case study: a system tracking circular objects;
- Operating Systems review;
- Another case study: the Spectre vulnerability
- Tasks, threads and interprocess communication;
- Real-time scheduling: definitions, cyclic executive, utilization based scheduling, response time analysis, priority inheritance;
- Data Acquisition techniques;
- GPU and FPGA and ZINQ architectures in real-time applications

References:

[1]I C Bertolotti, G Manduchi. Real-Time Embedded Systems. Open Source Operating Systems Perspective. CRC Press, 2012

[2]G C Buttazzo. Hard Real-Time Computing Systems. Predictable Scheduling Algorithms and Applications. Springer 2005.

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basic knowledge of Operating System concepts.

Examination and grading: Each student will develop a survey report based on one or several articles related to the material covered in class and referring to some field of application for real-time systems.

[back to **Summary**]

IE 22. Bayesian Machine Learning

Course Area: Information Engineering

Credits: 5

Instructor: Prof. Giorgio Maria Di Nunzio

e-mail: dinunzio@dei.unipd.it

Aim: The course will introduce fundamental topics in Bayesian reasoning and how they apply to machine learning problems. In this course, we will present pros and cons of Bayesian approaches and we will develop a graphical tool to analyse the assumptions of these approaches in classical machine learning problems such as classification and regression.

Topics:

- Introduction of classical machine learning problems.
  - Mathematical framework
  - Supervised and unsupervised learning
- Bayesian decision theory
  - Two-category classification
  - Minimum-error-rate classification
  - Bayes decision theory
  - Decision surfaces
- Estimation
  - Maximum Likelihood Estimation
  - Expectation Maximization
  - Maximum A Posteriori
  - Bayesian approach
- Graphical models
  - Bayesian networks
  - Two-dimensional visualization
- Evaluation
  - Measures of accuracy

References:

[1] J. Kruschke, Doing Bayesian Data Analysis: A Tutorial Introduction With R and Bugs, Academic Press 2010

[2] Christopher M. Bishop, Pattern Recognition and Machine Learning (Information Science and Statistics), Springer 2007

[3] Richard O. Duda, Peter E. Hart, David G. Stork, Pattern Classification (2nd Edition), Wiley-Interscience, 2000

[back to **Summary**]

[4] Yaser S. Abu-Mostafa, Malik Magdon-Ismail, Hsuan-Tien Lin, Learning from Data, AMLBook, 2012 (supporting material available at http://amlbook.com/support.html)

[5] David J. C. MacKay, Information Theory, Inference and Learning Algorithms, Cambridge University Press, 2003 (freely available and supporting material at http://www.inference.phy.cam.ac.uk/mackay/

[6] David Barber, Bayesian Reasoning and Machine Learning, Cambridge University Press, 2012 (freely available at http://web4.cs.ucl.ac.uk/staff/D.Barber/pmwiki/pmwiki.php?n=

[7] Kevin P. Murphy, Machine Learning: A Probabilistic Perspective, MIT Press, 2012 (supporting material http://www.cs.ubc.ca/ murphyk/MLbook/)

[8] Richard McElreath, Statistical Rethinking, CRC Presso, 2015 (supporting material https://xcelab.net/rm/statistical-rethinking/)

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basics of Probability Theory. Basics of R Programming.

Examination and grading: Homework assignments and final project.

IE 23. Learning from Networks

Course Area: Information Engineering

Credits: 5 (10 lectures, 2 hours each)

Instructor: Prof. Fabio Vandin

e-mail: fabio.vandin@unipd.it

Aim: The course introduces the area of learning from networks, including algorithms, theoretical foundations, and applications. The course will start with a short introduction to the area, continue with fthe undamental approaches, including traditional feature extraction based approaches and network embeddings, and move to more recent approaches, including graph neural networks. The course will cover (semi-)supervised (e.g., graph convolutional networks) and unsupervised (e.g., clustering) techniques for learning from networks.

Topics:

- Feature extraction based approaches
    - Topological features
    - Similarity based methods
    - Feature selection on attributed networks
- Network motifs
    - Mining network motifs
    - Motifs in temporal networks
- Network embeddings
    - Structure-preserving and property-preserving embeddings
    - Embeddings of attributed networks
- Graph neural networks
    - Graph convolutional networks
    - Inductive representation learning on large graphs
- Network clustering
    - Higher-order graph clustering
    - Clustering uncertain networks

References:

[1] He, X., Cai, D., & Niyogi, P. (2006). Laplacian score for feature selection. In *Advances in neural information processing systems* (pp. 507-514).

[2] Li, J., Wu, L., Zaïane, O. R., & Liu, H. (2017). Toward personalized relational learning. In *Proceedings of the 2017 SIAM International Conference on Data Mining* (pp. 444-452).

[back to **Summary**]

[3] Paranjape, A., Benson, A. R., & Leskovec, J. (2017). Motifs in temporal networks. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining* (pp. 601-610).

[4] Perozzi, Bryan, Rami Al-Rfou, and Steven Skiena (2014). "Deepwalk: Online learning of social representations." *Proceedings of the 20th ACM SIGKDD international conference on Knowledge Discovery and Data Mining*.

[5] Huang, X., Song, Q., Yang, F., & Hu, X. (2019). Large-Scale Heterogeneous Feature Embedding. In *AAAI Conference on Artificial Intelligence*.

[6] TN Kipf, M Welling (2017). Semi-supervised classification with graph convolutional networks. *5th International Conference on Learning Representations*.

[7] Yin, H., Benson, A. R., Leskovec, J., & Gleich, D. F. (2017). Local higher-order graph clustering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 555-564).

Schedule and room: please, see Class Schedule

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Course requirements: Basics of Probability. Basics of Machine Learning. Knowledge of (at least) one programming language.

Examination and grading: Presentation of a paper in front of the instructor and the other class attendees, and final project.

IE 24. Advanced topics in scientific and parallel programming with practical application to the CAPRI HPC infrastructure

Course Area: Information Engineering

Credits: 5

Instructor: Giacomo Baruzzo and Michele Schimd, Department of Information Engineering, University of Padova

e-mail: giacomo.baruzzo@unipd.it, michele.schimd@unipd.it

Aim: Provide basic skills for working on remote servers, using/developing parallel software and deploying it on a containerized computer server. The course gives basic introduction to modern computer architecture and to the most important parallel programming paradigms: Multi-threading, OpenMP, MPI and Cuda with examples (mostly Python). The course covers basic tools to access and to interact with remote servers, to manage remote resources, and to manage jobs. The course introduces principles of software containerization from the perspective of users, providing practical examples of Docker and Singularity. The concepts discussed are applied to simple case of studies involving writing and/or running parallel programs using the CAPRI HPC infrastructure (256 cores, 6TB shared RAM and 2 GPU Nvidia P100) recently acquired by the University of Padova for research activities.

Topics:

1. How to use a computing server (application to CAPRI)
   a. Connecting to the server and managing files (bash, ssh, scp, tunnelling, git, personal disk quotas, shared and temporary filesystems)
   b. Compiling, running, and saving results (choosing compiler, batch/interactive run, I/O and the redirecting of outputs)
   c. Job scheduling (slurm; writing a job; running, stopping and querying status of a job)
   d. The CAPRI queuing system and policy (CAPRI hardware and architecture; access to CAPRI and projects; execution queue; how to choose queue)
2. Containerization (singularity)
   a. Overview of containerization (definition of containers and container daemon; singularity and docker software; containers vs virtual machines; advantages: re-usability and reproducibility, flexibility, efficiency; disadvantages: learning curve)
   b. Using container that have already been defined (running, stopping, and resuming containers; containers options and flags)
   c. Defining new containers (new containers from scratch; extending existing containers)
   d. Sharing containers and the container repository (browsing and adding a container to the repository; guidelines for creating and documenting containers to be shared)
3. Multi-process/Parallel Programming
   a. Introduction to parallel programming models and architectures (basic definitions; shared vs distributed memory architectures; threading: CPU and GPU; shared memory programming; message passing programming; performance metrics)

[back to **Summary**]

  b. Parallel programming languages and frameworks (multi-threading; OpenMP; MPI; Cuda)
4. Hands on example (a simple parallel software for data analysis / machine learning; a simple parallel software for numerical analysis; students' proposals)

References:

- Eijkhout, V. (2013). Introduction to High Performance Scientific Computing. Lulu. com.
- Grama, A., Kumar, V., Gupta, A., & Karypis, G. (2003). Introduction to parallel computing. Pearson Education.
- Parhami, B. (2006). Introduction to parallel processing: algorithms and architectures. Springer Science & Business Media.
- Ad-hoc material by Lecturers

Schedule and room: please, see Class Schedule

Course requirements: Basics usage of tools for run/develop of scientific software (preferable unix platforms)

Enrollment: students must enroll in the course using the Enrollment Form on the PhD Program eLearning platform (requires SSO authentication).

Examination and grading: Each student must produce a small parallel and containerized software (either predefined or custom built container) related to her/his research field. Each student can either a) write a simple parallel software with one of the programming paradigm presented during the course using a language of choice or b) choose a (possibly parallel) software typically used in the research activity. Containerized software must run on the CAPRI server.

Alphabetical List of Course Instructors

[back to **Summary**]