MASTER OF SCIENCE
IN ENGINEERING

Module Description, available in: EN

Approximation algorithms

General Information

Number of ECTS Credits

3

Module code

FTP_ApprAlg

Valid for academic year

2020-21

Last modification

2019-12-08

Coordinator of the module

Fabrizio Grandoni (SUPSI, fabrizio.grandoni@supsi.ch)

Explanations regarding the language definitions for each location:

- Instruction is given in the language defined below for each location/each time the module is held.
- Documentation is available in the languages defined below. Where documents are in several languages, the percentage distribution is shown (100% = all the documentation).
- The examination is available 100% in the languages shown for each location/each time it is held.

	Lausanne			**Lugano**	**Zurich**		
Instruction				**X** E 100%			
Documentation				**X** E 100%			
Examination				**X** E 100%			

Module Category

FTP Fundamental theoretical principles

Lessons

2 lecture periods and 1 tutorial period per week

Entry level competences

Prerequisites, previous knowledge

Basics of theoretical analysis of algorithms.

Brief course description of module objectives and content

An algorithm is typically called efficient if its worst-case running time is polynomial in the size of the input. This course will focus on a huge and practically relevant family of problems, namely NP-hard ones, for which (most likely) no efficient algorithm exists. This family includes fundamental problems in computational biology, network design, systems, computer vision, data mining, online markets, etc.

The first goal of this course it to learn how to identify NP-hard problems.

For a given NP-hard optimization problem it might still be possible to compute efficiently a feasible solution whose cost is (in the worst-case) within a small multiplicative factor (approximation factor) from the optimum: this is the aim of approximation algorithms. The second goal of this course is to learn how to design accurate approximation algorithms, and how to (theoretically) bound their approximation factor.

Aims, content, methods

Learning objectives and competencies to be acquired

The main goal of this course is to learn how to identify NP-hard problems, and how to design and (theoretically) analyze approximation algorithms for fundamental NP-hard optimization problems.

Module content with weighting of different components

A. Complexity Classes: polynomial Reductions; NP-completeness and NP-hardness.

B. NP-hard problems: SAT and Max-SAT; Vertex/Set Cover; Steiner Tree and generalizations; TSP; Max Cut; Knapsack and Bin Packing; k-Center and clustering; Scheduling; Facility Location; Item Pricing; etc.

C. Approximation Algorithms:

1. Basic notions: approximation factor; hardness of approximation.
2. Basic techniques: greedy; local search; randomization; dynamic programming.

LP-based techniques: randomized rounding; primal-dual; iterative rounding.

Teaching and learning methods

Interactive lectures both for theory and exercises.

Literature

- V. V. Vazirani. Approximation Algorithms. Springer.

- D. P. Williamson, D. B. Shmoys. The Design of Approximation Algorithms. Cambridge University Press.

Assessment

Certification requirements

Module does not use certification requirements

Basic principle for exams

As a rule, all standard final exams are conducted in written form. For resit exams, lecturers will communicate the exam format (written/oral) together with the exam schedule.

Standard final exam for a module and written resit exam

Kind of exam

Written exam

Duration of exam

120 minutes

Permissible aids

No aids permitted

Special case: Resit exam as oral exam

Kind of exam

Oral exam

Duration of exam

30 minutes

Permissible aids

No aids permitted