1 Online lectures

Class sessions:

- Lecturer: Fabricio Oliveira (fabricio.oliveira@aalto.fi);
- Session info: Thu, 14:15h - 16:00h, via Zoom (please check MyCourses Forum for link);

Exercise sessions:

- Teaching Assistant: Nikita Belyak (nikita.belyak@aalto.fi);
- Session info: Thu, 16:15h - 18:00h, on Zulip (please check MyCourses Forum for platform link).

2 Course description

Mathematical optimisation (nonlinear optimisation, in its most general form) is a powerful framework in which one seeks to find variable values within a domain that maximise (or minimise) the value of a given function. Using the analogy that variables represent decisions or parameters to be defined and the function is a measure of performance, one can use that framework to support decision making in a wide range of applications, from planning industrial chemical plants to training models that learn from data.

In this course, the student will learn the basic optimisation theory behind the main numerical algorithms available and how they can be applied to solve optimisation problems. At the end of the course, it is expected that the student will be capable of analysing the main characteristics of an optimisation problem and decide what is the most suitable method to be employed for its solution.

3 Learning outcomes

Upon completing this course, the student should be able to

- understand how several important problems arising from diverse fields can be cast and solved as nonlinear optimisation problems;
- know the main techniques for solving nonlinear optimisation problems and how to apply them in practice;
- know how to use optimisation software for implementing and solving nonlinear optimisation problems.

4 Teaching methods

The course will be taught by a composition of the following methods:

- online lectures;
- guided self-study;
- theoretical and computational exercises;
- project assignment and feedback.

The lectures will be given live via Zoom sessions. All lectures will also be recorded and made available on MyCourses. For the exercise sessions and to address general questions about the course content or administration, we will use Zulip. Zulip is the standard chat platform in the Department of Mathematics and Systems Analysis and offer the possibility for calls and using Latex on the messages.

As preparation for the lectures (Thursdays at 14.15h-16.00h), the students will be requested to study the lecture notes (about 10 pages per lecture) beforehand and formulate questions to be submitted before (via Zulip) or during the lecture sessions (via Zoom chat or asking live).

The exercise sessions (Thursdays at 16.15h-18.00h) will happen as self-study sessions, in which the students are expected to study the exercises, by watching the recorded tutorials and trying to solve selected exercises on their own. Videos presenting the solutions of the exercises will be made available as video recordings, but we strongly suggest the students to watch them after trying to do the exercises themselves.

The time slot of the exercise sessions will work as office hours for the students to clarify questions related to content, homework and assignments. Whenever possible but during weekdays only, we will monitor the course's Zulip chat to answers any questions. However, for timely support, we recommend that the students join the exercise sessions on Zulip and ask questions there (either via text or via a call on Zulip).

5 Assessment

The final grade of the course is composed of three components:

H: 4 homework assignments;
$P1$: Project assignment 1;
$P2$: Project assignment 2;

Each component will be graded individually in a scale of 0-100. The final grade FG will be calculated as

$$FG = 0.5 \times H + 0.25 \times P1 + 0.25 \times P2.$$

The conversion scale for to the 1-5 scale is as follows.

1-5	0-100
Fail	0-50
1	51-60
2	61-70
3	71-80
4	81-90
5	91-100

Table 1: Conversion from 0-100 to 1-5 scale

5.1 Homework assignments

A total of 4 homework assignments will be handed out. Each homework is worth 25 points, adding to a total of 100 points. The homework for each week will be available on Monday from MyCourses and will have as deadline the Monday evening of the following fortnight (two weeks). The submission of the solutions must be made through MyCourses. Homework submissions after the deadlines will have a 5 point discount plus extra 5 points per each day (24h) after the deadline.

The homework will be composed by theoretical and computational exercises. The computational skills required to solve the exercises will be introduced in the exercise tutorials, but it is expected that the student learn and practise the language on its own. Supporting material for that will be provided. The programming language that will be used in this course is Julia (julialang.org).

5.2 Project assignments

The students will be requested to develop guided projects on two distinct topics to be provided. The objective of the projects is to use the acquired knowledge in nonlinear optimisation in practice and discuss related technical aspects.

The projects can be conducted individually or in pairs. Each project will comprise an implementation using Julia that address the requirements of the projects and a 3000-4000 word (approximately 5-7 pages) report. Each report will be graded by the course lecture.

6 Course material

Main study material: lecture notes, lecture slides, exercise tutorials, homework assignments, course book.

The lecture material is mostly based on the main course book *M. Bazaraa et al., Nonlinear Programming, Wiley-Interscience, 2006.*

7 Course schedule

A tentative schedule for the course is given. Content of each class may be adapted according to the pace of the classes.

Week	Lecture	Content
37	1	Admin. and introduction
38	2	Topology - Convex sets
39	3	Analysis - Convex functions
40	4	Optimality conditions I
41	5	Unconstrained optimisation methods I
42	6	Unconstrained optimisation methods II
43	–	*Break between Periods I and II*
44	7	Optimality conditions II
45	8	Lagrangian duality
46	9	Constrained optimisation methods I
47	10	Constrained optimisation methods II
48	11	Constrained optimisation methods III
49	12	Computational and practical points

Table 2: Schedule of classes