Aalto University

Non-Linear Optimization

MS-E2122

Lecturer Info

Fernando H. C. Dias

Office Hrs: Wednesday 14:00-15:00

Otakaari 1, Y214

fernando.dias@aalto.fi

Course Info

Thursday

14:15-16:00

213a - 213a

Study Session Info

Wednesday

12:15-14:00

DELOITTE - U119 (Period I), U6 KONECRANES - U149 (Period 2)

TA Info

Topias Terho

topias.terho@aalto.fi

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 performance measure, one can use that framework to support decision-making in a wide range of applications (even beyond mathematics and engineering), from planning industrial chemical plants to training models that learn from data and bioinformatic tools.

In this course, the student will learn the basic optimisation theory behind the main numerical algorithms available and how they can be applied to 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 deciding the most suitable method for its solution.

Material

Required Texts

Lectures Notes, Lecture Slides and Charts. All are available on "MyCourse" Page.

Text Book

M. Bazaraa et al., Nonlinear Programming, Wiley-Interscience, 2006.

Additional Material

Any extra material suggested throughout lectures will be available on "MyCourse" Page.

Learning Objectives

With completion of this course, each student should be capable of:

- Apply the theory and the algorithm from the course as a tool to solve real and more complex problems involving non-linear optimization;
- Familiarise themselves with classical algorithms for constrained and unconstrained optimization;
- Familiarise themselves with classical non-linear problems;
- Learn the difference between approximation, heuristics and greedy algorithms and know where to apply them wisely;
- Learn convexity and understand how important and vital this concept is to optimization algorithms ;
- know the main techniques for modelling and solving nonlinear optimisation problems and how to apply them in practice;
- know how to use optimisation software for implementing and solving nonlinear optimisation problems;
- Understand the limitation of non-linear optimization and the concept of branching.

Teaching Methods

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

- lectures;
- theoretical and computational exercises;
- project assignment and feedback.

The lectures will be in person as well as the exercise sessions. 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 offers the possibility for calls and using Latex on messages. Zulip link will be added to the course MyCourses page before the start of the course.

As preparation for the lectures (Thursday 14:15-16:00pm), the students will be encouraged to study and familiarise themselves with the lecture notes (about 10 pages per lecture) beforehand and formulate questions to be submitted before (via Zulip) or during the lectures. Lecture notes and charts are available on "MyCourse".

The exercise sessions (Wednesday 12:15-14:00pm) will happen as self-study sessions, in which the students are expected to study the exercises and by the end, the teaching assistant will help and provide solutions to such exercises. In addition, whenever possible, due to time constraints, additional support on homework will also be provided.

The time slot of the exercise sessions will work as office hours for the students to clarify questions related to content and assignments. We will monitor the course's Zulip chat whenever possible to answer any questions. However, for timely support, we recommend that the students join the exercise sessions on Zulip and ask questions there.

Remark: Reception hour:
Lecturer: at 13:00 - 14:00 in room Y214 (Otakaari 1). Please confirm the appointment by contacting the lecturer via email first.

Assessment

The final grade of the course is composed of five homework assignments. Each assignment will be graded individually in a scale of 0-100. The final grade FG will be calculated as a sum of all homework points normalized on a scale of 0-100.

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

1-5	0-100
Fail	0-49
1	50-59
2	60-69
3	70-79
4	80-89
5	90-100

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

Attendance is not mandatory in this course . However, for each class atended each student will be awarded +0.5 points per lectures (leading to a total of 6 extra points).

Homework assignments

A total of 5 homework assignments will be handed out. Combined, the homework assignments will count for 100 points. The homework will be available on specific Mondays (see schedule below) from "MyCourse" and will have a deadline of three to four weeks (depending on the assignment - see schedule below). The submission of the solutions must be made through "MyCourse". Homework submissions after the deadlines will have a 3-point discount plus an extra 1 point each day (24 hours) after the deadline.

The homework will be composed of theoretical and computational exercises. The computational skills required to solve the exercises will be introduced in the exercise tutorials, but the student is expected to learn and practise the language independently. Supporting material for that will be provided. The programming language that will be used in this course is Python (python.org) (preferably) or Julia (julialang.org). Submissions using other languages will be allowed (but preferably use one of the suggested).

Workload estimate

The following breakdown shows an estimate of the total workload of the course. Recall that each ECT credit is equivalent to 28 hours of work; thus, 5 ECTs are equivalent to 140 hours.

- Pre-Lecture/Exercise Session preparations require 2 hour per week (as recommendation) (24 hours in total)
- Exercise Sessions require 2 hours per week (24 hours in total);
- Lectures requires 2 hours per week (24 hours in total);
- Homework assignments require 10 hours per assignment (60 hours in total);
- The course requires 130 hours (5 ECT).

Deadline Extension Policy

Extension for the deadline for assignments and/or projects will only be allowed for students who have preemptively contacted the lecturer or teaching assistant *before the due date*. In case of extraordinary events, the student should notify either the lecturer or teaching assistant afterwards; then, a special exception will be discussed for each case.

Equality, Diversity and Inclusivity Statement

I consider this classroom to be a place where you will be treated with respect, and I welcome individuals of all ages, backgrounds, beliefs, ethnicities, genders, gender identities, gender expressions, national origins, religious affiliations, sexual orientations, ability - and other visible and non-visible differences. All class members are expected to contribute to a respectful, welcoming and inclusive environment for every other class member.

Accommodations for Students with Special Requirements

If you are a student with learning needs that require special accommodation, contact me as soon as possible to discuss how to improve your participation in this course. Please e-mail me as soon as possible to schedule a time to discuss your learning needs.

Academic Integrity

Research and Academic Integrity are central to the ideals of this course. Students are expected to be independently familiar with them and recognize that their work in the course will be their original work that truthfully represents the time and effort applied. Use of any AI-generated text or code is prohibited.

Lectures Schedule

Part I: Fundaments to Convexity

Week 1	*Introduction*
	Admin. and Course Introduction
	Optimization Basics & Convexity I
Week 2	*Convexity*:
	Convexity II
	Sets and Functions Analysis

Part II: Unconstrained Optimization

Week 3	*Optimality Conditions*:
	First and Second Order Optimality Conditions
Week 4	*First Order Methods*:
	Gradient and Coordinate Descent
	Conjugated Gradient
	Stochastic Gradient
Week 5	*Second Order Methods*
	Newton's method
	BFGS and DFP
	Convergence and Conditioning

Part III: Constrained Optimization

Week 6	*KKT and FJ*
	Constrained Optimization
	Karush-Kuhn-Tucker conditions
	Fritz-Johnson conditions
Week 7	*Lagrangian Duality*
	Weak Duality
	Strong Duality
	Properties of Lagrangian Duality
Week 8	*Lagrangian Methods*
	Augmented Lagrangian Method
	Alternating Lagrangian Method

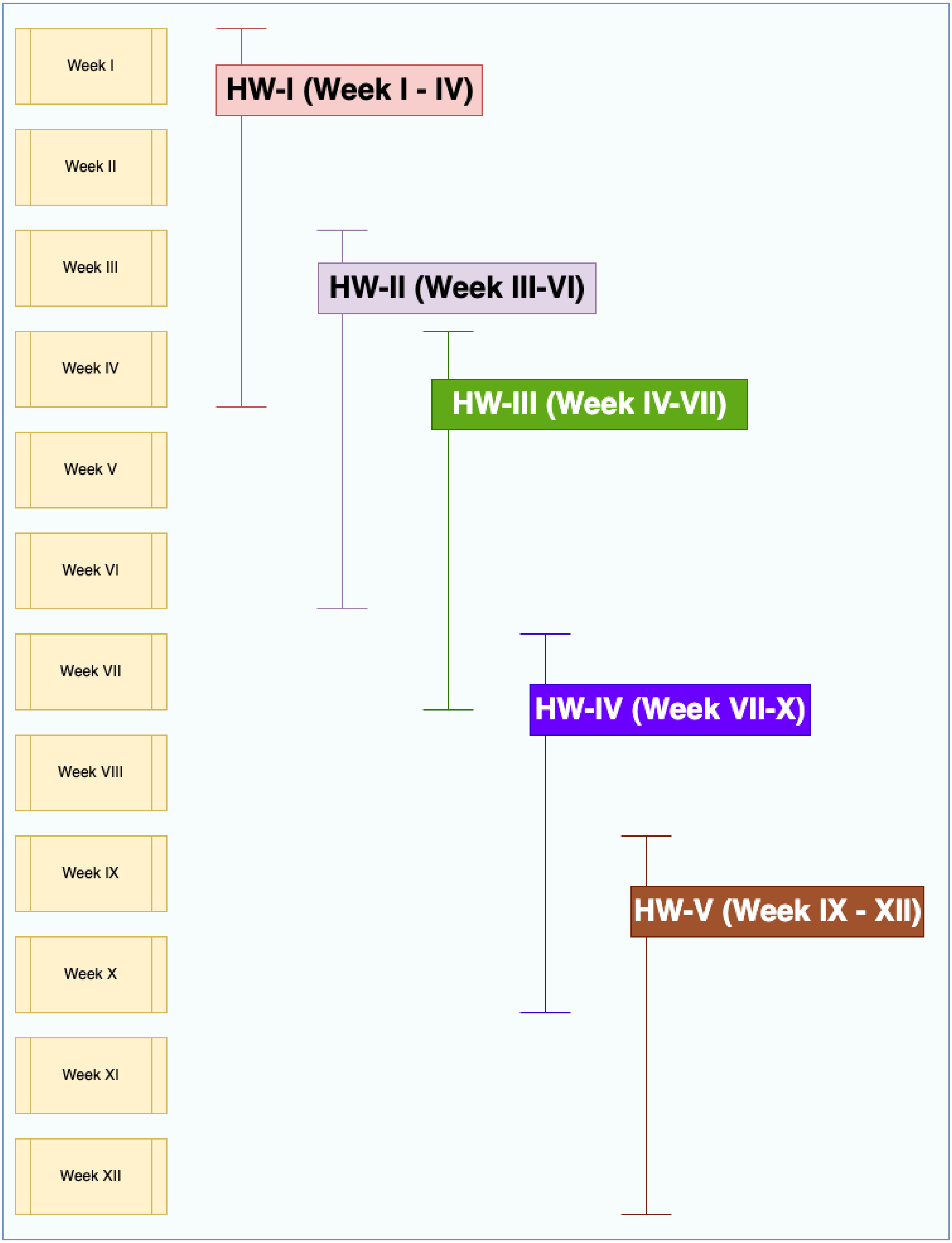
Week 9	*Barrier Methods*
	Barrier Functions
	Barrier Methods
	Interior Points
Week 10	*Primal Methods*
	Frank-Wolf
	SQP
Week 11	*General Methods*
	Spatial Branching
	Future of Non-Linear Optimization

Part IV: Real World

Week 12	*Guest*
	Guest Lecture
	(subject to change based on guest availability)

Teaching Sessions Schedule

Week 1	Convexity I	Exercises session - Publication of HW-I
Week 2	Convexity II	Exercises session
Week 3	Optimality Conditions	Exercises session - Publication of HW-II
Week 4	First Order Methods	Exercise session - Deadline of HW-I and Publication of HW-III
Week 5	Second Order Methods	Exercise session
Week 6	KKT and FJ	Exercise session - Deadline of HW-II
EXAM WEEK		
Week 7	Lagrangian Methods	Exercise session - Deadline of HW-III, Publication of HW-IV
Week 8	Barrier Methods (Barrier & Conics)	Exercise session
Week 9	Primal Methods	Exercise session - Publication of HW-V
Week 10	General Methods	Exercise session - Deadline of HW-IV
Week 11	Project Support	Exercise session
Week 12	Project Support	Exercise session - Deadline of HW-V



Assessments - Schedule &Deadline