1 Lectures

Class sessions:

- Lecturer: Fabricio Oliveira (fabricio.oliveira@aalto.fi);
- Session info: Tue & Thu, 13:00h - 16:00h (4 sessions)

2 Course description

Stochastic programming and robust optimisation are powerful mathematical programming approaches for modelling problems with uncertain data. However, often these models suffer from an increase in computational requirements to an extent it may hinder their practical deployment. One way to remediate computational tractability issues is by means of employing decomposition, an algorithmic strategy that allows the problem to be broken into parts that can be more efficiently solved, potentially in a parallel manner.

In this course, we will introduce the main decomposition techniques available to tackle challenges related to computational requirements. We will discuss their main strengths and what are the key challenges for their successful application. At the end of the course, it is expected that the student will develop a general understanding of the main methods that can be applied to decompose stochastic programming and robust optimisation problems, including some practical experience in coding such methods.

3 Teaching methods

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

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

4 Prerequisites

The student is expected to have a basic background in linear programming. Specifically, it is important that the student has familiarity with concepts such as:

- formulations of (mixed-integer) linear programming problems, such as knowing what are decision variables and constraints, objective function, and the role of linearity in the primal/dual simplex method;
- familiarity with the notion of convexity, and related concepts such as convex hulls, convex combinations, and so forth;
- a general understanding of the notion of Lagrangian duality in the context of linear programming, including the formulation of duals;
- some experience with programming in Julia and using packages such as JuMP and optimisation solvers (e.g., Gurobi).

5 Learning outcomes

Upon completing this course, the student should

- have a grasp the basics of the main techniques available for decomposing stochastic programming and robust optimisation problems;
- understand of the main difference between the methods, and their strengths and weaknesses;
- develop a critical thinking approach for considering more sophisticated state-of-the-art methods.

6 Teaching methods

The lectures will be given in person only. Following each lecture, there will be a guided exercise session. The exercise is computational using Jupyter notebooks. If possible, it is recommended that the students follow the exercise in real-time with their own personal computers.

In preparation for the computational exercise, a test notebook will be given. The students are responsible for making sure that they can run the notebook correctly with appropriate versions of the packages. We will use the most recent versions of all packages required.

In preparation for the lectures (Thursdays at 14.15h-16.00h), the students will be requested to study the lecture notes (slides) beforehand and formulate questions to be asked during the lecture.

7 Assessment

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

H: 4 short homework assignments;
P: Project assignment;

Each component will be graded individually on a scale of 0-10. The final grade FG will be calculated as

$$FG = 0.4 \times H + 0.6 \times P$$

7.1 Homework assignments

A total of 4 homework assignments will be handed out. The homework will be composed of computational exercises complementing those done in class. The computational skills required to solve the exercises will be introduced in the exercise tutorials and will be based on the exercises done in class.

7.2 Project assignments

The students will be requested to prepare a 20-minute presentation on a scientific article of their choice. The presentation must be recorded (audio over slides) and submitted to a Dropbox folder that will be provided. The presentation will be peer-graded, meaning that each student will be requested to provide feedback on the other presentations as well.

8 Course material

Main study material: lecture notes, lecture slides, exercise tutorials, and additional material (scientific papers) provided in class.

9 Course schedule

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

Week	Lecture	Content
1	1	Benders decomposition
1	2	Column-and-constraint generation
2	3	Dual decomposition
2	4	ADMM and Progressive hedging

Table 1: Class topics and schedule