1 Course Outline

The course is on Numeric optimization and its applications. The course will be focused on Linear Algebra in scientific computing, Convex optimization, including mixed-integer. The applications in robotics and control will have a focus.

2 Grading Policy

Grading criteria:

- A: 85-100 points
- B: 70-84 points
- C: 50-69 points
- D: less than 50 points

Credit distribution:

- 20 points can be awarded for HW / lab 1
- 20 points can be awarded for HW / lab 2
- 15 points can be awarded for HW / lab 3
- 5 points can be awarded for in-class activities
- 40 points are awarded for the exam

2.1 Late Submission Policy

Points might be deducted if the work is submitted after deadline

3 Course Structure

The course structure is given in the table 1.

4 Course Outline

4.1 Knowledge Areas (in terms of application)

- Robotics
- Automation
- Design (mechanical, electrical)

- Control
- Data analysis
- Computational geometry

4.2 Course Delivery

The lectures are given every week, followed by practical sessions (labs). During the sessions, students are required to develop their own code based on the knowledge acquired during lectures and the self-study.

4.3 Prerequisite courses

- Strong prerequisites: Linear Algebra.
- Weak prerequisites: Calculus, Control Theory.
- Required background knowledge: Python (alternatively Matlab or any other language suitable to work with linear algebra-heavy problems)

4.4 Expected Learning outcomes

The course will provide an opportunity for participants to:

- Understand and learn to use Convex Programming (CP): Quadratic programming, Linear Programming, SOCP, SDP (LMI), as well as Mixed-integer CP.
- Learn how to use convex optimization solvers, especially CVX.
- Being able to implement optimization on practical problems, especially in robotics, control, computational geometry.

4.5 Expected acquired core competencies

- Developing efficient linear algebra-based code for control and robotics applications.
- Developing solutions using numeric optimization, especially convex optimization.
- Optimization-based methods in Robotics.

4.6 Reference Materials

- Annotated slides
- Online materials
- Educational videos

4.7 Computer Resources

Students will need to run computer experiments on a laptop and/or on lab computers.

4.8 Laboratory Exercises

There are a series of labs and electronic handouts prepared for the course.

4.9 Laboratory Resources

Students will be required to use and modify a software tool written in Python which run on multiple platforms (Linux, Microsoft Windows, and Mac OS). The tool requires freely available software libraries.

4.10 Cooperation Policy and Quotations

We encourage intensive discussion and collaboration in this class. You should feel free to discuss all aspects of the class with classmates and work with them to complete your assignments and project report. However, if you are working together, you must provide details of your contribution and that of others.

Table 1: Course Structure

Lecture 1-2	Introduction. Least Squares Linear Algebra and Four Fundamental Subspaces (Null space, Row space, Column and Left null spaces). Projectors, SVD decomposition.	Individual assignment
Lecture 3-6	Quadratically constrained quadratic programming Second order cone programming (SOCP). Quadratic Programming. Convex Domains.	Individual assignment
Lecture 7-10	Semidefinite programming (SDP). Linear matrix inequalities in Control. Ellipsoids. Barrier functions, center of constraints. Sensitiviy analysis.	
Lecture 11-13	Minimax. Mixed-integer convex programming. Relaxation. Big M method. Example in path planning.	