Chinese University of Hong Kong, Shenzhen
School of Science and Engineering

CIE/DDA 6010 OPTIMIZATION THEORY AND ALGORITHMS

Information Sheet v.1
Autumn 2019

Lecturer

Professor DRAPER, Stark
Email: starkdraper@cuhk.edu.cn (or, alternately) stark.draper@utoronto.ca
Office: Dao Yuan 515
Office hours: TBA

Teaching Assistant

HUANG, Zhipeng
Email: huangzhipeng@cuhk.edu.cn
Office: TBA
Office hours: TBA

Welcome to CIE/DDA6010

This class will introduce you to the fundamental theory and models of optimization as well as the geometry that underlies them. We start our discussion with geometry: recalling and generalizing linear algebraic concepts. We then move on to optimization. Presentation of applications is woven throughout. We will draw examples from diverse areas of the engineering and natural sciences. The material covered in this course will prove of interest to students from all areas of engineering, from the computer sciences and, more generally, from disciplines wherein mathematical structure and the use of numerical data is of central importance.

Prerequisites

The main prior courses that we will be building on are vector calculus and linear algebra.

Course Text

Optimization Models, by G. Calafiore and L. El Ghaoui, Cambridge Univ. Press, 2014.

Lectures

Monday: 16:30-18:00, Cheng Dao Building, room 207.
Wednesday: 16:30-18:00, Cheng Dao Building, room 207.
Highly interactive lectures are the most useful and engaging. Questions are welcomed and encouraged.

Office Hours

Office hours are an informal time to ask questions and learn from me, the course staff, and from each other. It's better to discover early in the semester the usefulness of office hours, rather than waiting until the end of the term. Please come by!

Course Website

All course handouts will be posted on Blackboard. Please make sure you can access. Also please let me know if you have not received course emailings (sent via Blackboard). We need to have everyone on that list.

Problem Sets

The course problem sets are designed to improve your understanding of the course material. They will include both written and numerical problems. Problem sets may include optional problems. You are expected to do all assigned (i.e., non-optional) problems. In designing the exams I will assume you have worked and/or studied the solutions to all problems, including the optional ones.

The problem sets are a crucial part of the learning experience. They are designed to illustrate course concepts and material. Without working through and, perhaps, struggling at length with the problems, you will not develop as deep a facility with the concepts developed in class. Invariably this will have a major impact on the depth of your understanding and your final grade.

- COLLABORATION: Collaboration with one or two classmates on problem sets is encouraged. However, each student must *individually* write up their own solutions. Please note on your solutions the names of your collaborator(s).

- PROFESSIONAL PREPARATION OF YOUR WORK—NAMED AND STAPLED: Please prepare your work in a professional manner. As well as writing legibly, and answering the questions *in order*, you should (i) note your name at the top of your problem set, and (ii) staple your problem set. Deductions *will be assessed* for non-professional preparation. I will not bring a stapler to lecture – don't ask.

- HANDING IN PROBLEM SETS: Problem sets must be handed in during the class in which they are due. Solutions will be distributed at the end of class and therefore late problem sets will *not* be accepted. If you are planning to travel during a week in which a problem set is due, arrange to have a colleague drop it off, or email the TA a scanned version *before* the end of the class in which the problem set is due.

- NUMERICAL PORTION: For all problems for which you are asked to develop code, include a print-out of your code along with your solution to the problem. Inclusion of your code demonstrates that you completed the problem in a way that sole inclusion of, e.g., plotted results, does not. Deductions will be assessed for missing code. While problems are designed for use with Matlab, e.g., through inclusion of sample snippets of code, you are welcome to use whatever programming language you prefer.

The mathematical tools and perspectives we develop in the class find application in an enormous range of technologies. One objective of the problems sets is to introduce you to some of these diverse applications and to introduce you to standard numerical toolboxes (e.g., `CVX`) that are used in some of these technologies. All problem sets will have numerical parts through which you will be introduced to some of these applications. Examples of applications include PageRank, latent semantic indexing, Eigenfaces, optimal control under various norms, CAT scanning, portfolio optimization, sparse image coding.

Exams

There will be two exams in the course. The first will be held sometime in the middle of the semester. The second will be held at the end of the semester. You will be allowed to bring a "cheat" sheet to each exam. To the first exam you may bring **two** 8.5×11 sheets of nodes (both sides). To the second exam you may bring **three** 8.5×11 sheets of nodes (both sides). Otherwise both exams will be closed-book and closed-notes.

Neither exam has yet been scheduled. Details will be announced in class and via Blackboard.

Project Information

An important aspect of the course is the course project. Ideally you will identify a project that is related to your research and technical interests. Please get an early start discussing possible course project topics with your supervisor.

Projects will be done in teams, the size of teams will be determined by the ultimate class size. I expect most projects to have a significant numerical component. I also expect in your presentation and report to hear some interesting new observations, e.g., a comparison of different approaches in two different papers or a new problem formulation, theoretical development. There will be an in-person presentation and a written report. Clarify of exposition will count towards the grade. Further details will be announced subsequently.

Course Grade (CIE 6010)

The weighting of the problem sets, the midterm exam, and the final exam will be:

Problem sets:	25%
Project:	15%
Midterm:	20%
Final exam:	40%

Course Grade (DDA 6010)

The weighting of the problem sets, the midterm exam, and the final exam will be:

Problem sets:	25%
Project:	20%
Midterm:	20%
Final exam:	35%

Reference Texts

There are a number of reference texts for this course. The text by Axler and the two texts by Boyd and Vandenberghe are available in electronic form on their respective authors' websites (in the case of the Axler text in a condensed form).

1. *Introduction to linear algebra* by Gilbert Strang, Wellesley-Cambridge Press. Now in its 6th edition. This is a classic text. It's the text I used to learn linear algebra. Strang is a great teacher. See the book's website for lots of online resources: `http://math.mit.edu/~gs/linearalgebra/`.

2. *Linear Algebra Done Right*, by Sheldon Axler, Springer Press, 2015. This is a text for a "second course" in linear algebra. It develops linear algebra for finite-dimensional vector spaces in a fully mathematical manner. It is a great and easy-to-read textbook that I highly recommend. Results we don't prove in the course are proved here. If you are motivated to understand linear algebra even more fundamentally, you will like this text. If you do decide to get a hard-copy you may want to look for the second edition (the 2015 edition is the third). While somewhat less material is covered in the second edition, the formatting is less rich and therefore (in my opinion) is nicer and easier to read. Please note that an electronic version of a condensed version the third edition of this text available on the author's website.

3. *Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares* by Stephen Boyd and Lieven Vandenberghe, Cambridge University Press, 2018. An introductory linear algebra text. Good for review. Touches on some topics that may not have been covered in your linear algebra course, e.g., non-Euclidean vector norms. We may assign some problems from this textbook. They are accessible via the online version.

4. *Convex Optimization*, by Stephen Boyd and Lieven Vandenberghe, Cambridge University Press, 2004. This text is used in many graduate optimization courses. An electronic version of this text is available on the authors' websites.

To delve futher into the course material, the following are classic references.

5. *Nonlinear Programming*, by Dimitri P. Bertsekas, Athena Scientific, 1999. This is a widely used alternative to Boyd and Vandenberghe's *Convex Optimization* text.

6. *Optimization by Vector Space Methods*, David G. Luenberger, Wiley, 1969. This classic focuses on optimization over infinite-dimensional vector spaces.

7. *Convex Analysis*, R. Tyrell Rockafellar, Princeton University press, 1997. This reprint of the original is the place to go to see the story of convexity from a more mathematical perspective. Beautiful proofs.

Topics, Tentative Syllabus, and Readings

A list of course topics is provided below. Specific sections that we do *not* cover will be posted to help focus your reading. Sections not posted not to read means you are expected not not to read them.

- Introduction (Chapter 1)
- Linear algebra:
  - Vector and functions (Chapter 2)
  - Matrices (Chapter 3)
  - Symmetric matrices (Chapter 4)
  - Singular value decomposition (Chapter 5)
- Convex optimization models:
  - Linear equations and least squares (Chapter 6)
  - Linear, quadratic, and geometric models (Chapter 9)
  - Convexity (Chapter 8)
  - Introduction to algorithms (chapter 12)
- Further topics*
  - Matrix algorithms (Chapter 7)
  - Second-order cone and robust models (Chapter 10)
  - Semidefinite models (Chapter 11)
  - Applications (Chapters 13-16)

*Depending on how we progress through the term we will optionally cover materials from these chapters, sometimes in lecture, sometimes in the problem sets.