COMPUTATIONAL OPTIMIZATION - DNSC 8392

School of Business – The George Washington University

INSTRUCTOR: DR. MIGUEL A. LEJEUNE

Office: FUNGER HALL, 406

Phone: 202-994-6576

E-mail: mlejeune@gwu.edu

Office Hours: On appointment

Time: Wednesday, 3:30PM – 6PM

Place: Duques 262

OVERVIEW

This course is about the description, design, and programming of efficient computational methods for large-scale optimization problems. Examples of such problems abound and can be found in network design, transportation, supply chain, production and distribution, data mining, health care, treatment optimization, vaccination strategies, military operations, vehicle or crew routing and scheduling, facility location, financial asset allocation, telecommunications, capacity allocation problems. These problems can take multiple forms (single- vs. multi-objective; deterministic vs. stochastic; continuous vs. integer; convex vs. non-convex, etc.). Despite their continuous improvement, academic and commercial optimization software are not always sufficient to satisfy the ever increasing demand from industry for solving large instances of such problems.

In order to develop successful methods for large-scale challenging problems, it is essential to effectively integrate relaxation, reformulation, decomposition methods as well as search algorithms that exploit structural properties of the problems of interest. Such approaches can yield provably optimal solutions or good approximate solutions to large-scale problems in practice.

The course will provide students guidance to efficiently formulate optimization problems and to derive algorithms for the solution of large-scale and data-driven optimization problems that are hard to solve. The course will acquaint the students with the software, optimization solvers, and programming languages used by professionals to code and model industry-size optimization problems. Applications to practical problems will be introduced throughout the course.

The course starts with a typology of the different types of mathematical optimization/programming problems. A review of modeling aspects in mathematical optimization and basic algorithmic methods will follow. Modeling and implementations of solution methods with the algebraic modelling language AMPL and optimization solvers (e.g., Cplex, Gurobi, Mosek, Bonmin, Ipopt, Couenne, Cbc, Knitro, Snopt) will be detailed.

The second main part of the course will be devoted to nonlinear optimization with a focus on convex programming. Convex programming computational methods (reformulation, relaxation, linearization) will be implemented with AMPL, Matlab, and the Disciplined Convex Programming Matlab package CVX.

The third part will be devoted to some mixed-integer nonlinear programming, stochastic programming, and distributionally robust optimization problems. Relaxations, decomposition, and inner approximation methods will be reviewed and their implementation using AMPL and Matlab will be presented. Illustrations with C++ and the Mixed-Integer Linear and Nonlinear Matlab packages MILANO and YALMIP might also be covered.

The course will be highly interactive with intensive computer practice. The students will be presented with various problems to work on during the semester.

PREREQUISITE

This course is suitable for students with some knowledge in optimization and some basic programming experience.

OBJECTIVES

The main objectives of this course are to:

- develop the ability to formulate good mathematical optimization formulations for large-scale problems.
- detect and exploit the properties of mathematical optimization problems to design computationally efficient methods for large-scale data-driven optimization problems.

- learn key concepts and the basic theory underlying the numerical solution of mathematical optimization problems.
- develop programming and coding skills to implement reformulation and algorithmic methods and to interact with off-the-shelf optimization solvers.
- incorporate large-scale optimization methods into your research.
- get acquainted with the software, programming and modeling languages, and methods used by professionals to solve large-scale optimization problems. In that respect,
  - the AMPL algebraic modeling language will allow you to:
    - formulate compactly large-scale optimization models.
    - interact with optimization solvers.
    - develop algorithms complementing optimization solvers and allowing for the efficient solution of large-scale problems.
    - build dynamic link libraries (DLL) and to employ them in the optimization process.
  - the optimization solvers - commercial solvers, such as Cplex, Gurobi, and Mosek, and open-source solvers, such as Cbc, Bcp, Ipopt, Couenne, Bonmin - will allow you to solve complex optimization problems and to complement your specialized algorithms.
  - the Matlab software package will allow you to:
    - compactly formulate large-scale optimization models.
    - interact with optimization solvers, in particular solvers customized for classes of nonlinear continuous optimization problems (e.g., Sedumi).
    - efficiently execute vector and matrix computations and decompositions.
    - use specialized optimization packages, such as CVX (and maybe Yalmip and MILANO), that are particularly effective for particular families of optimization problems.
  - the C++ (tentative) programming language will allow you to:
    - formulate optimization problems.
    - allow for an optimal interaction with the callable libraries of optimization solvers.

COURSE MATERIAL

Class notes and copies of papers will be distributed.

Suggested textbooks are:

- Boyd S., Vandenberghe L. 2004. *Convex Optimization.* Cambridge University Press. Available online:

  https://web.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf.

- Davis T.A. 2011. *MATLAB. A Primer.* 8th Edition. Taylor & Francis, CRC Press, Boca Raton, FL.
- Fourer R., Gay D.M., Kernighan B.W. 2002. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press / Brooks / Cole Publishing Company. Available online: http://www.ampl.com/BOOK/download.html
- IBM. 2012. IBM ILOG AMPL Version 12.2 - User's Guide.
- Lee J. 2017. *A First Course in Linear Optimization.* Third Edition (Version 3.00), Reex Press, 2013-17. Available online: https://github.com/jon77lee/JLee_LinearOptimizationBook/blob/master/JLee.3.0.pdf

  Required software and programming languages:

- Algebraic Modelling Language AMPL (see Blackboard).
- Matlab.
- Commercial optimization solvers: Cplex, Gurobi, Knitro, Minos, Snopt (see Blackboard).
- Open-source optimization solvers: Bonmin, Ipopt, Couenne, Cbc (see Blackboard).
- CVX: Matlab Software Package for Disciplined Convex Programming (see Blackboard).

  Other software and programming languages that might be discussed:

- C++;
- Yalmip: Matlab Software Package for convex optimization;
- Milano: Matlab Software Package for Mixed-Integer Linear and Nonlinear Optimization.

TENTATIVE SCHEDULE

The list of topics is tentative and is subject to update:

- Lecture 1:
  - Typology of Optimization Problems.
  - Overview of Optimization Techniques and Solvers.
  - Review of Linear Optimization Problems.

- Lectures 2-3:
  - Mixed-Integer Linear Optimization Problems.
  - Modeling Logical and Combinatorial Constraints.

- Lecture 3: Modeling with AMPL.

- Lecture 4:
  - Convexity Concepts.
  - Convex Programming.

- Lecture 5:
  - Lagrangean Relaxations.
  - Duality for Nonlinear Problems.
  - Optimality Conditions.
  - Optimization with Matlab.

- Lecture 6:
  - Quadratic Programming.
  - Quadratic Duality.
  - Optimization with Matlab.

- Lectures 7-8:
  - Conic Programming.
    - Second-Order Cone Programming.
    - Semi-Definite Programming.
    - Conic Duality.

- Lecture 8:
  - Other Conic Optimization Problems.
  - SOCP-representability.
  - Geometric Programming.
- Lecture 9:
  - Optimization with CVX Matlab Package.
  - Mixed-Integer Ninlinear Programming.
- Lectures 10-11: Mixed-Integer Ninlinear Programming.
- Lecture 12: Mathematical Programming Formulations of Risk Metrics: Financial Applications.
- Lectures 13-14:
  - Decomposition Methods.
  - Dual Dynamic Programming.
  - Stochastic Dual Dynamic Programming.
  - Presentations.

GRADING

The course grade will be based on homework assignments, class participation, students' presentations, and an exam (or research project subject to the instructor's approval). Tentatively, 35% of the grade will be based on the homework assignments, 35% will be based on the project or final exam, 20% on presentation(s), and 10% will be based on participation. The objective of the assignments is to develop confidence in efficiently formulating optimization problems, and designing and implementing optimization methods using software packages. Computational projects will require programming.

ASSIGNMENT OF CREDIT HOUR POLICY

Students will spend 2.5 hours per week in class. Out of class, they will spend an average (per week) of 12 hours for readings, homework and computational assignments, and projects. Over the course of the semester, students will spend 35 hours in instructional time and 168 hours preparing for class. Instructional time includes discussions and activities in class.

ACADEMIC INTEGRITY

The code of academic integrity applies to all courses in the George Washington School of Business. Please become familiar with the code. All students are expected to maintain the highest level of academic integrity throughout the course of the semester. Please note that acts of academic dishonesty during the course will be prosecuted and harsh penalties may be sought for such acts. Students are responsible for knowing what acts constitute academic dishonesty. The code may be found at: http://www.gwu.edu/~ntegrity/code.html

Special Considerations:

If a student has a special need, University policy states that the student must co-ordinate with the Office for Disability Services and present the course instructor with the appropriate documentation detailing the fair accommodations for the student. This policy is intended to ensure fairness for all students and privacy for the student with special needs. If you have a special need, please do not wait until after an exam or assignment to present the instructor with evidence of your need as consideration may not be given retroactively. The instructor ensures your privacy will be protected when accommodating special needs.

The University administration has accepted a resolution of the Faculty Senate regarding accommodations of religiously observant students and faculty. The requirements of this resolution state that students must notify faculty during the first week of the semester of their intention to be absent from class on their day(s) of religious observance. Faculty member will extend to these students the courtesy of absence without penalty on such occasions, including permission to make up examinations.