CO 250: Introduction to Optimization

Fall 2013 – Tentative Course Outline

minor updates to this outline may occur in the first 2 weeks of classes

1 Instructors & lectures

<table>
<thead>
<tr>
<th>Section</th>
<th>Instructors</th>
<th>Office</th>
<th>Phone</th>
<th>Lecture room</th>
<th>Lecture time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bill Cook</td>
<td>MC 6314</td>
<td>x38148</td>
<td>QNC 1502</td>
<td>10:30-11:20 MWF</td>
</tr>
<tr>
<td>2</td>
<td>Joseph Cheriyan</td>
<td>MC 5034</td>
<td>x35591</td>
<td>MC 4020</td>
<td>10:00-11:20 TuTh</td>
</tr>
</tbody>
</table>

2 Tutorials & office hours (optional)

<table>
<thead>
<tr>
<th>Room</th>
<th>Time</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office hour</td>
<td>MC 6314</td>
<td>Wed. 1:00–2:30</td>
</tr>
<tr>
<td>Office hour</td>
<td>MC 5034</td>
<td>Tue. 11:45–1:15</td>
</tr>
<tr>
<td>Tutorial</td>
<td>B2 350</td>
<td>Mon. 4:30–5:20</td>
</tr>
<tr>
<td>Tutorial</td>
<td>B2 350</td>
<td>Tue. 4:30–5:20</td>
</tr>
<tr>
<td>Tutorial</td>
<td>B2 350</td>
<td>Wed. 4:30–5:20</td>
</tr>
</tbody>
</table>

Tutorial of Oct. 29 (Tue) will be held in PHY 313.
Tutorials of Sep. 18, Oct. 16, Oct. 30 (Wed) will be held in PHY 235.

Note: B2 is the Biology 2 building; it is next to the QNC building, and there are links connecting MC and QNC, as well as QNC and B2.

The tutorial sessions and office hours will be the appropriate forum for all questions pertaining to the course material and assignments. Students are encouraged to attend. If you prefer to have your questions answered in a tutorial style meeting and would like to hear the questions of like-minded students and the TA’s answers to others’ questions, then attend the tutorials. If you need a one-on-one meeting, then choose the regular office hours. Note that you may attend any of the office hours or tutorials, regardless of the section in which you are enrolled.

3 Midterm exam and Final exam

A midterm exam will be held on Oct. 23 (Wed.), 7:00–8:50 (rooms: DC 1350, DC 1351); there will be an alternate session on the same day, 5:00–6:50, in MC 4045. Those wishing to write in the alternate session must register with the instructor of their section (by Oct. 9) and are required to stay in the exam hall till 6:50pm. Please contact your instructor (by Oct. 9) if you have conflicts with both sessions.

The final exam will be scheduled by the Registrar’s Office, and will be held in December.

Missed exams will count as zero unless suitable medical documentation is provided.

4 Overview

Suppose that the owner of a factory wants to maximize its production for the next 30 days. There is a limit on the resources available. Resources may include, raw materials, labor, machine capacities, etc. This is an example of an optimization problem. The function that we are trying to maximize
is the **objective function**, and the conditions imposed by the available resources are the constraints of the problem. Optimization problems are classified according to the type of objective function and the type of constraints.

The simplest models are *linear programs* where both the constraints and the objective functions are linear. Even though this may appear at a first glance to be overly restrictive, linear programming algorithms are used widely across most branches of industry. Indeed, a recent survey of Fortune 500 companies shows that 85% of all respondents use such algorithms in their operations. It is not hard however, to imagine applications for which fractional variable values are not desirable. For instance a variable may indicate the number of employees to hire, or a variable may be restricted to values 0 or 1 to indicate one of two possible options (e.g., build a factory in Waterloo or don’t). In these cases we would like to add the condition that some variables in our linear program take integer values only. These models are known as *integer programs*. Finally, in certain instances, such as portfolio optimization (in financial mathematics), the natural way of formulating the optimization problem may require the use of non-linear constraints, or a non-linear objective function.

In the first part of CO250, we will illustrate these various models with examples that arise from real problems. The later part of the course addresses the subject of how to solve the aforementioned problems. The *Simplex* algorithm to solve linear programs will be discussed in some detail and general-purpose integer programming techniques such as *cutting planes* will also be described. These algorithms while guaranteed to terminate, may in the worst case (and often do in practice) take a prohibitively long time. No fast general algorithm is known for integer programs (and none is believed to exist), however, there are efficient algorithms for many important special cases. An indispensable tool for the design of such fast algorithms is the theory of *duality*, which will be a main focus of this course. This discussion will lead us to *non-linear convex optimization* problems.

We will supplement some of the fundamental topics by giving a geometric perspective; for example, we will show a geometric view of the simplex algorithm, and we will present a geometric illustration of the complementary slackness conditions.

### 5 Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–2</td>
<td>Formulations (linear, integer, and non-linear)</td>
</tr>
<tr>
<td>3–5</td>
<td>Solving linear programs</td>
</tr>
<tr>
<td>5–9</td>
<td>Duality theory and applications to algorithms</td>
</tr>
<tr>
<td>10–11</td>
<td>Integer programming</td>
</tr>
<tr>
<td>11–12</td>
<td>Non-linear convex optimization</td>
</tr>
</tbody>
</table>

### 6 Objectives

After completing the course, students will be expected to master the following tasks and concepts:

- Formulate simple real life problems as linear, integer, or continuous (non-linear) optimization problems.
- Carry out hand computations for simple instances of various algorithms such as simplex.
- Formulate the dual of various linear programs.
- Explain the role of duality theory in the design of algorithms for combinatorial optimization.
• Reproduce the main proofs from the course, as well as independently prove simple related concepts.

• Explain the geometric interpretation of the various algorithms covered.

Note regarding algorithms: while it is necessary to know how to carry out simple computations by hand, it is not sufficient. We expect students to have a good understanding of why algorithms return correct answers, and the rationale behind the design of each algorithm.

7 Grades

We will use the following weights to compute your final grade in the course:

• Assignments 20%

• Midterm 30%

• Final 50%

7.1 Assignments

There will be eight to ten assignments to be handed in. We will post assignments on Friday, and they are due the following week on Friday, 10:00a.m., in the drop boxes located on the 4th floor of MC (outside the tutorial center); see the course webpage for further info. Late assignments will not be graded and a mark of zero will be assigned. The assignments and their solutions will be posted on the course webpage.

When computing the “assignments grade” of a student, we will ignore the worst assignment of the student, e.g., a student who gets 100% on 8 assignments, 10% on one assignment, and 0% on one assignment will get 90% (or 18/20) for the assignments grade.

While it is acceptable for students to discuss the course material and the assignments, you are expected to write solutions to assignments on your own. For example, copying or paraphrasing a solution from some fellow student or old solutions from previous offerings of related courses qualifies as cheating and we will instruct the TA’s to actively look for suspicious similarities and evidence of academic offenses when grading.

All students found to be cheating will be given a mark of zero on the assignment (where the mark of zero will not be ignored as the lowest assignment). In addition, all academic offenses are reported to the Associate Dean for Undergraduate Studies and are recorded in the student’s file (this may lead to further, more severe consequences). Furthermore, cheating students will receive a 5% penalty on their final mark; i.e., a mark of 74% would be reduced to 69%.

If you have any complaints about the marking of assignments, then you should first check your solutions against the posted solutions. After that, if you see any marking error, then you should return your assignment submission to the instructor of your section within one week, and with written notes on all the marking errors; please write the notes on a new sheet and attach it to your assignment submission.

8 Course webpage

We will be using LEARN this term for our course webpage. Log on using your username and password (same as your UW email account). All course related information including additional
reading material, assignments, and announcements will be posted there. It is the responsibility of each student to check the course webpage regularly.

9 INC grade

A grade of INC (incomplete) will be only awarded to students who cannot write the final exam for reasons acceptable to the instructor, such as a medical certificate by a recognized medical professional. In addition such students need to be in good standing prior to the final exam. To be in good standing a student must

- submit and pass at least 6 of the assignments,
- write and pass the midterm exam, and
- attend classes regularly.

10 Academic integrity & students with disabilities

The following is reproduced from http://www.uwaterloo.ca/accountability/documents/courseoutlinestmts.pdf.

Academic Integrity  In order to maintain a culture of academic integrity, members of the University of Waterloo community are expected to promote honesty, trust, fairness, respect and responsibility. Check http://www.uwaterloo.ca/academicintegrity/ for more information.

Grievance  A student who believes that a decision affecting some aspect of his/her university life has been unfair or unreasonable may have grounds for initiating a grievance. Read Policy 70, Student Petitions and Grievances, Section 4, http://www.adm.uwaterloo.ca/infosec/Policies/policy70.htm. When in doubt please be certain to contact the department’s administrative assistant who will provide further assistance.

Discipline  A student is expected to know what constitutes academic integrity to avoid committing academic offenses and to take responsibility for his/her actions. A student who is unsure whether an action constitutes an offense, or who needs help in learning how to avoid offenses (e.g., plagiarism, cheating) or about rules for group work/collaboration should seek guidance from the course professor, academic advisor, or the undergraduate associate dean. For information on categories of offenses and types of penalties, students should refer to Policy 71, Student Discipline, http://www.adm.uwaterloo.ca/infosec/Policies/policy71.htm. For typical penalties check Guidelines for the Assessment of Penalties, http://www.adm.uwaterloo.ca/infosec/guidelines/penaltyguidelines.htm.

Appeals  A decision made or penalty imposed under Policy 70, Student Petitions and Grievances (other than a petition) or Policy 71, Student Discipline may be appealed if there is a ground. A student who believes he/she has a ground for an appeal should refer to Policy 72, Student Appeals, http://www.adm.uwaterloo.ca/infosec/Policies/policy72.htm.
Note for students with disabilities  The Office for Persons with Disabilities (OPD), located in Needles Hall, Room 1132, collaborates with all academic departments to arrange appropriate accommodations for students with disabilities without compromising the academic integrity of the curriculum. If you require academic accommodations to lessen the impact of your disability, please register with the OPD at the beginning of each academic term.