IOE510/MATH561/OMS518: Linear Programming I

- **Lectures:** Mondays and Wednesdays, 9:00-10:30 AM, IOE 1680
- **Instructor:** Prof. Marina A. Epelman
- **GSI:** Katharina Ley
- **Office hours:** See CTools for most current info
- Other course materials, including assignments, lecture topics and these slides are available on CTools
- **Required background:** Rigorous course in linear algebra (if in doubt, review section 1.5 of the textbook and see instructor); a level of mathematical maturity.

Course Logistics

- Weekly homeworks, including some computational work
  - Due before class or in Katharina’s mailbox by 12:30 PM
  - We apologize, but we cannot collect homeworks or answer questions after the class — gotta run
  - No late submissions, but lowest score dropped
- Two in-class midterms (mid-October and mid-November)
- Final exam Monday, 12/20, 4:00-6:00 PM

**Individual work policy summary:** you are allowed, indeed, encouraged to work together on homework *conceptualizing the problems*. However each student is individually responsible for expressing their answers in their own terms, writing their own solutions and code to be submitted. Also you may not acquire, read, or otherwise utilize answers from solutions handed out in previous terms. *Full explanation of these and other policies are in the syllabus.*
Course goals

- Discuss Linear Programming as a mathematical technique to model decision and optimization problems relevant in engineering, various industries and other applications, as well as methods for solving the resulting models and interpret the solutions.
- Since Linear Programming bridges the fields of engineering and applied and pure mathematics, this course will cover, and ask you to perform in homeworks and exams,
  - development of mathematical models of real problems, both on paper and through computer modeling languages,
  - rigorous proofs of mathematical results,
  - development and analysis of algorithmic methods for solving linear programming problems,
  - presentation of your ideas and solutions in precise language, correctly using appropriate technical terms and other terminology.

Informal course outline

- Mathematical Optimization and Linear Programming (LP): what is it and what is it good for? (modeling examples)
- Geometry of LP problems
- Simplex method for solving LPs
- Duality theory of LP
- Sensitivity analysis
- Network flow problems
- Intro to path-following Interior Point Methods
- Intro to Integer Linear Programming
Optimization problems

An optimization problem is a

- Set of decisions to make
- A single quantitative criterion to compare decisions
- Rules governing interactions of decisions

Example: the Diet Problem

Given a collection of foods (e.g., all items in your favorite supermarket),

- Decide how much of each item to eat in a given day,
- ... with the objective of choosing the cheapest daily diet,
- ... while meeting all the rules/constraints governing nutritional requirements (not too much fat, not too many calories, sufficient amount of protein, etc.)

Diet problem: some history

- Posed by Stigler in 1945
  - Solved heuristically
- Dantzig 1963, 1990
  - Solved optimally in 1947 using simplex method
  - 120 man-hours
- Garille and Gass 2001
  - Solved using current dietary guidelines and prices
  - Solution (“optimal diet”): 1.31 cups wheat flour, 1.32 cups rolled oats, 16 oz. milk, 3.86 tbsp peanut butter, 7.28 tbsp lard, 0.0108 oz. beef, 1.77 bananas, 0.0824 oranges, 0.707 cups cabbage, 0.314 carrots, 0.387 potatoes, 0.53 cups pork and beans
Diet problem: lessons learned

- Decision making in a complex system:
  - Very large number of choices to consider
  - Many rules to satisfy
  - Decisions interact

- The diet problem is a concrete illustration of a wide array of ideas and concepts:
  - Need good sources of data to pose the problem (how much protein in 1 cup of wheat flour?)
  - Distinguish optimization problem from problem instance:
    - Diet problem: “choose how much of available items to eat in order to minimize cost while satisfying nutritional constraints.”
    - An instance of the diet problem: depends on what items are sold at your store, how much they cost, and what your specific nutritional needs are.
  - Allows for mathematical insight:
    - Isn’t it surprising that out of all food items available, only 12 are in the optimal diet?

Mathematical Program

A Mathematical Program\(^1\) is a mathematical representation, or model, of an optimization problem in which decisions that need to be made are quantitative.

- Decisions ⇔ Decision variables
- Comparison criterion ⇔ Objective function
- Rules governing interactions ⇔ Constraints

\(^1\)Note the outdated use of the term “program” — it means “plan,” or “schedule,” not “computer program” — the term was coined in the 1940s.
Mathematical Programming model of the Diet Problem

- Suppose we have 5 food items (numbered 1 through 5) to choose from, and 3 nutritional guidelines to meet (fat, calories, protein).
- Decision variables: $x_1, x_2, x_3, x_4, x_5$ — how many ounces of each item to eat in a day
- Need the following data:
  - Price, in $ per ounce, of each item: $c_1, c_2, c_3, c_4, c_5$
  - Amount of fat, in grams per ounce, in each item: $f_1, \ldots, f_5$
  - Amount of calories\(^2\), per ounce, and protein, in grams per ounce, in each item: $k_1, \ldots, k_5$ and $p_1, \ldots, p_5$, respectively
  - Maximal amounts (for calories and fat) and minimal amount (for protein) required in a day: $K$ calories, $F$ grams, and $P$ grams, respectively

\(^2\)kilo-calories, if you are talking to a real nutritionist

Model of the Diet Problem — continued

\[
\begin{align*}
\text{minimize} & \quad c_1 x_1 + c_2 x_2 + c_3 x_3 + c_4 x_4 + c_5 x_5 \\
\text{subject to} & \quad f_1 x_1 + f_2 x_2 + f_3 x_3 + f_4 x_4 + f_5 x_5 \leq F \\
& \quad k_1 x_1 + \cdots + k_5 x_5 \leq K \\
& \quad \sum_{j=1}^{5} p_j x_j \geq P \\
& \quad x_1 \geq 0, \ldots, x_5 \geq 0
\end{align*}
\]

(Objective f-n) (Fat constraint) (Calorie constraint) (Protein constraint) (Nonnegativity constraints)
Instance of the Diet Problem

The oddball diet: data

<table>
<thead>
<tr>
<th>Food</th>
<th>Price, $/oz</th>
<th>Fat, g/oz</th>
<th>Cal., kcal/oz</th>
<th>Protein, g/oz</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chocolate</td>
<td>0.75</td>
<td>5</td>
<td>250</td>
<td>1</td>
</tr>
<tr>
<td>Special K</td>
<td>0.55</td>
<td>0</td>
<td>90</td>
<td>2</td>
</tr>
<tr>
<td>Liver</td>
<td>1.5</td>
<td>1</td>
<td>300</td>
<td>3</td>
</tr>
<tr>
<td>OJ</td>
<td>0.45</td>
<td>0</td>
<td>75</td>
<td>4</td>
</tr>
<tr>
<td>Pizza</td>
<td>1.25</td>
<td>10</td>
<td>400</td>
<td>5</td>
</tr>
</tbody>
</table>

Limit

≤ 30
≤ 2400
≥ 10

Instance of the Diet Problem

The oddball diet: model instance

minimize \(0.75x_1 + 0.55x_2 + 1.5x_3 + 0.45x_4 + 1.25x_5\)
subject to
\[5x_1 + x_3 + 10x_5 \leq 30\]
\[250x_1 + 90x_2 + 300x_3 + 75x_4 + 400x_5 \leq 2400\]
\[x_1 + 2x_2 + 3x_3 + 4x_4 + 5x_5 \geq 10\]
\[x_1 \geq 0, \ldots, x_5 \geq 0\]
Linear Programs (LPs)

- Notice that the model of the Diet Problem had the following properties:
  - Continuous variables
  - Linear objective function
  - Constraints are linear inequalities
- Such mathematical programs are called Linear Programming problems, or Linear Programs, or LPs, and they are the subject of this course.
- Despite seeming simplicity, widely applicable in the real world
- Have beautiful mathematical properties, which allowed the discovery of fast algorithms for solving even very large instances
- Provide the foundation for analysis and solution methods for more general mathematical programs (i.e., non-linear, or ones with discrete variables)

Areas of applications of LP models and methods

Efficient resource allocation, military operations planning, production and inventory planning, capacity expansion, manufacturing process design, staff scheduling, location planning, traffic routing, supply chain management, economic game theory, airline crew and plane scheduling, telecommunication capacity allocation and network design, medical treatment planning, image reconstruction, publishing (typesetting), finance (asset allocation), mathematics (as a proof technique and computational method), data analysis, pattern classification, optimal control, mechanical structure design, electromagnetic antennae design, etc.

Read sections 1.2-1.3 for examples!
Treatment planning in radiation oncology

- Over a million people in US are diagnosed with cancer each year; about half are treated with radiation therapy
- Radiation beams are directed at the patient from several different angles; as the beam passes through patient’s body, it delivers a dose of radiation to the cells it passes through
- Delivered doses of radiation damage the cells; cancerous cells are less likely to recover than healthy cells, thus, tumor is reduced
- Overall goal: deliver a high dose of radiation to the tumor while sparing the healthy tissue, and especially the critical structures, as much as possible.

Illustration
Obtaining problem data:

- A digital image of the patient is obtained via CT/MRI scans; tumor and surrounding critical structures are identified in the image; image is stored as a set of grid points $S$, indexed by $j \in S$, or pixels.
- Beam directions are selected; beams represented as a collection of beamlets indexed by $p = 1, \ldots, P$.
- Compute the dose delivery data: $A_{j,p}$ is the dose delivered to pixel $j$ by the beamlet $p$ with intensity of 1.
- If the beamlets are assigned nonnegative intensities $w_p$, $p = 1, \ldots, w_P$, the dose delivered to pixel $j$ is

$$D_j = \sum_{p=1}^{P} A_{j,p}w_p, \ j \in S.$$
**Idealized model**

**The problem:** Given the set of the tumor pixels $T$, set of critical structure pixels $C$, and the set of non-critical normal tissue pixels $N$ (that is, $S = T \cup C \cup N$), determine the intensity for each beamlet, $w_p$, $p = 1, \ldots, P$, such that the resulting dose delivered to the tumor pixels is at least $\gamma_T$, to the critical structure pixels – at most $\gamma_C$, and to the normal pixels – at most $\gamma_N$, and the total dose endured by the patient is minimized.

$$
\min_{w,D} \sum_{j \in S} D_j \\
\text{s.t.} \quad D_j = \sum_{p=1}^{P} A_{j,p} w_p, \quad j \in S
$$

$$
w_p \geq 0, \quad p = 1, \ldots, P
$$

$$
D_j \geq \gamma_T, \quad j \in T
$$

$$
D_j \leq \gamma_C, \quad j \in C
$$

$$
D_j \leq \gamma_N, \quad j \in N
$$