CO 370

Deterministic Operations Research Models

Instructor: Laurent Poirrier

lpoirrier@uwaterloo.ca

What is this course about?

- Theory and practice of **Operations Research (OR)**,
- **application** to real problems.

Operations Research:

$\rightarrow$ solve decision-making problems

using mathematical modeling and optimization.

Applications:

- military (logistics, supply chain)
- manufacturing (scheduling, lot sizing)
- transportation (vehicle routing, shortest path)
- telecommunications (flow problems, network design)
- financial engineering (portfolio management)
- ...

What will we learn?

- Modeling: **formulate** problem mathematically,
- Solving: find solution **algorithmically**.

Objective of the course:

solve practical problems on your own computer.

What will we learn?

- Modeling: **formulate** problem mathematically,
- Solving: find solution **algorithmically**.

Objective of the course:

solve practical problems on your own computer.

Course organisation

(see syllabus on `learn`)

Material covered:

- Linear optimization
- Integer optimization
- Decision-making under uncertainty
- Solving large-scale models.

Prerequisite: CO 250 or similar.

Grading

- 4 homework assignments 20%
- Midterm 30%
- Final 50%

Check `learn` / web page of the course!

Part I: Linear Optimization

Linear Programming (LP) is the problem of maximizing or minimizing a **linear function** subject to a finite number of **linear constraints**.

A **Linear Programming (LP)** is characterized by

- decision variables,
- a linear objective function,
- a (finite) number of linear constraints.

Example 1

A company produces NH_3 and NH_4Cl.

component	stock
N	50 kmol
H	180 kmol
Cl	40 kmol

product	revenue
NH_3	\$40 / kmol
NH_4Cl	\$50 / kmol

Plan production to maximize profit.

Example 1

$$
\begin{array}{lrcrcl}
\max & 40x_1 & + & 50x_2 & & \\
\text{s.t.} & x_1 & + & x_2 & \leq & 50 \\
& 3x_1 & + & 4x_2 & \leq & 180 \\
& & & x_2 & \leq & 40 \\
& x_1 & & & \geq & 0 \\
& & & x_2 & \geq & 0
\end{array}
$$

Example 1

```
@variable(model, x1)
@variable(model, x2)

@objective(model, Max,     40 * x1 + 50 * x2          )

@constraint(model, N,           x1 +       x2 <= 50   )
@constraint(model, H,       3 * x1 +  4 * x2 <= 180  )
@constraint(model, Cl,                     x2 <= 40   )
@constraint(model, NH3,         x1            >= 0    )
@constraint(model, NH4Cl,                  x2 >= 0    )
```

Example 1

```
julia> optimize!(model)

julia> objective_value(model)
2300.0

julia> value(x1)
20.000000000000004

julia> value(x2)
29.999999999999996
```