# Examples of Mathematical Programs

## (JSS30 Exercises Day01a)

# Classification: Mathematical Programming

| | Search-space $\mathbb{S}$ | Degree of nonlinearity |
|---|---|---|
| Linear Programming (LP) | $\mathbb{R}^d$ | linear |
| Quadratic Programming (QP)* | $\mathbb{R}^d$ | quadratic** |
| Nonlinear programming (NLP) | $\mathbb{R}^d$ | nonlinear |
| Integer programming (IP) | $\mathbb{Z}^d$ | arbitary |
| Integer linear programming (ILP)* | $\mathbb{Z}^d$ | linear |
| Mixed Integer Linear Programming (MILP) | $\mathbb{R}^d \times \mathbb{Z}^r$ | linear |
| Mixed Integer Nonlinear programming (MINLP) | $\mathbb{R}^d \times \mathbb{Z}^r$ | nonlinear |
| Continuous unconstrained optimization: $\mathbb{S} = \mathbb{R}^n$, $n_g = 0$ | | nonlinear |

*A QP is also an NLP; A ILP is also a IP.

A quadratic function is of the form:

$$c_0 + b_1 x_1 + \ldots + b_d x_n + a_{1,1} x_1 x_1 + a_{1,2} x_1 x_2 + \ldots + a_{d,d} x_d x_d$$

$$x^T A x$$

For the comprehensive authorative classification of INFORMS by Dantzig, see:
http://glossary.computing.society.informs.org/index.php?page=nature.html

# Multiobjective Mathematical Program

Let $x_1, \ldots, x_d$ denote $d$, $c_1, \ldots, c_n$, and $b_1, \ldots, b_q$ be defined as previous. A multiobjective mathematical programming problem (MOP) has the form:

$$\begin{array}{rcl} f_1(x_1, \ldots, x_d) & \rightarrow & \text{min} \\ & \vdots & \\ f_m(x_1, \ldots, x_d) & \rightarrow & \text{min} \end{array}$$

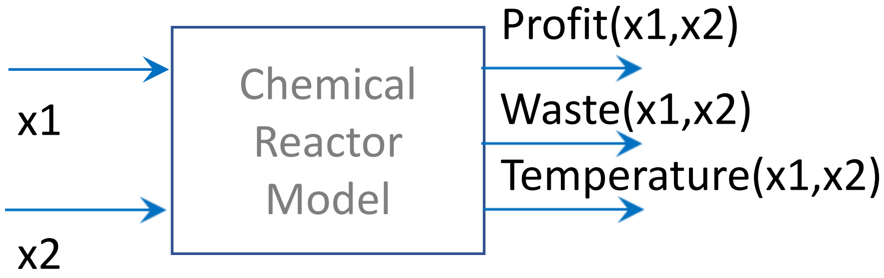
subject to

$$\begin{array}{rcl} g_1(x_1, \ldots, x_d) & \geq & c_1 \\ & \vdots & \\ g_n(x_1, \ldots, x_d) & \geq & c_n \\ h_1(x_1, \ldots, x_d) & = & b_1 \\ & \vdots & \\ h_q(x_1, \ldots, x_d) & = & b_q \end{array}$$

For $m > 1$ one can always add the term 'Multiobjective', e.g. Multiobjective LP, Multiobjective MIP, etc..

# Example 1: Mathematical Program for Reactor

Chemical Reactor




Profit to be maximized, while temperature and waste must not exceed certain thresholds.

How to formulate this as a mathematical program?

- Decision variables: Concentrations of educts: $x_1 = c_1/\left[\frac{g}{l}\right],\quad x_2 = c_2/\left[\frac{g}{l}\right]$
- Mathematical Program:

$$f(x_1, x_2) = \frac{Profit(x_1, x_2)}{[€]}$$

subject to

$$g_1(x_1, x_2) = \frac{Temp(x_1, x_2) - T_{max}}{[℃]} \leq 0$$

$$g_2(x_1, x_2) = \frac{Waste(x_1, x_2) - W_{max}}{\left[\frac{kg}{h}\right]} \leq 0$$

$$(x_1, x_2) \in [0,1] \times [0,1]$$

# Example 2: Constrained 0/1 Knapsack Problem

Picture: © Michael Emmerich (instructor)

The total value of the items in the knapsack (in [$]) should be maximized, while its total weight (in [kg]) should not exceed MAXWEIGHT. Here $v_i$ is the value of item i in [$] and $w_i$ is its weight in [kg]. i=1, …, d are indices of the items.

$$f_1(x_1, \ldots, x_d) = \sum_{i=1}^{d} \frac{v_i}{[\$]} x_i \rightarrow \max$$

$$g_1(x_1, \ldots, x_d) = \sum_{i=1}^{d} \frac{w_i}{[kg]} x_i - MAXWEIGHT \leq 0$$

$$x_i \in \{0,1\}, i = 1, \ldots, n$$

What is the role of the binary variables here?

What type of mathematical programming problem is this?

Can this also be formulated as a quadratic programming problem?

# Example: Multiobjective 0/1 Knapsack Problem

The total value of the items in the knapsack (in [$]) should be maximized, while its total weight (in [kg]) should be minimized.

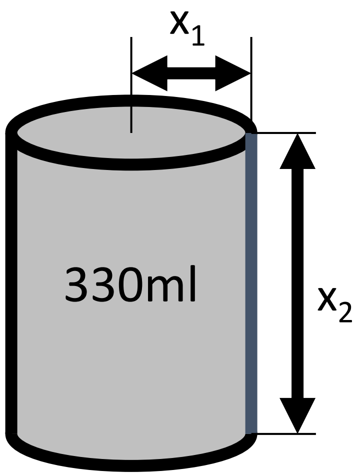
$$f_1(x_1, \ldots, x_d) = \sum_{i=1}^{d} \frac{v_i}{[\$]} x_i \rightarrow \max$$

$$f_2(\mathbf{x}) = \sum_{i=1}^{d} \frac{w_i}{[kg]} x_i \rightarrow \min$$

$$x_i \in \{0,1\}, i = 1, \ldots, n$$

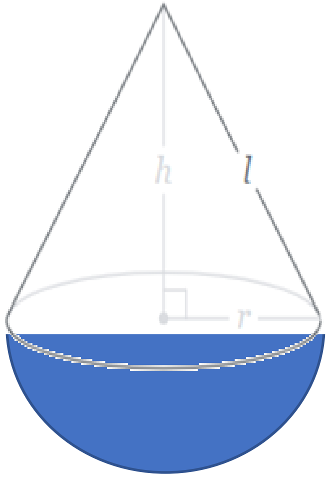
# Example: Equality Constraint for Tin Problem

Minimize the area of surface A for a cylinder that contains V = 330 ml sparkling juice! $x_1 = radius/[cm^2], x_2 = height/[cm^2]$




Problem sketch

$$f(\mathbf{x}) = 2\pi x_1 x_2 + 2\pi (x_1)^2 \rightarrow \min$$
$$h(\mathbf{x}) = 2\pi x_2\,(x_1)^2 = 330$$
$$\mathbf{x} \in \left[\binom{0}{0}, \binom{\infty}{\infty}\right] \subset \mathbb{R}$$




$$V = \pi r^2 \frac{h}{3}$$

$$A = \pi r\left(r + \sqrt{h^2 + r^2}\right)$$

Inside cone

Exercise: Formulate the problem for icecream cones as a mathematical programming problem! Surface of waffle to be 100ml Volume maximized (1/2 sphere+cone volume)

Some interesting research question: Find optimal shapes or given constraints on geometry.
For instance: Convex hull of N points with minimal surface and maximal volume.

# Example: Knapsack Problem with Cardinality Constraint

The total value of the items in the knapsack (in [$]) should be maximized, while its total weight (in [kg]) should be below MAXN and at most MAXN items can be chosen.


$$f_1(x_1,\dots,x_d) = \sum_{i=1}^{d} \frac{v_i}{[\$]} x_i \rightarrow \max$$

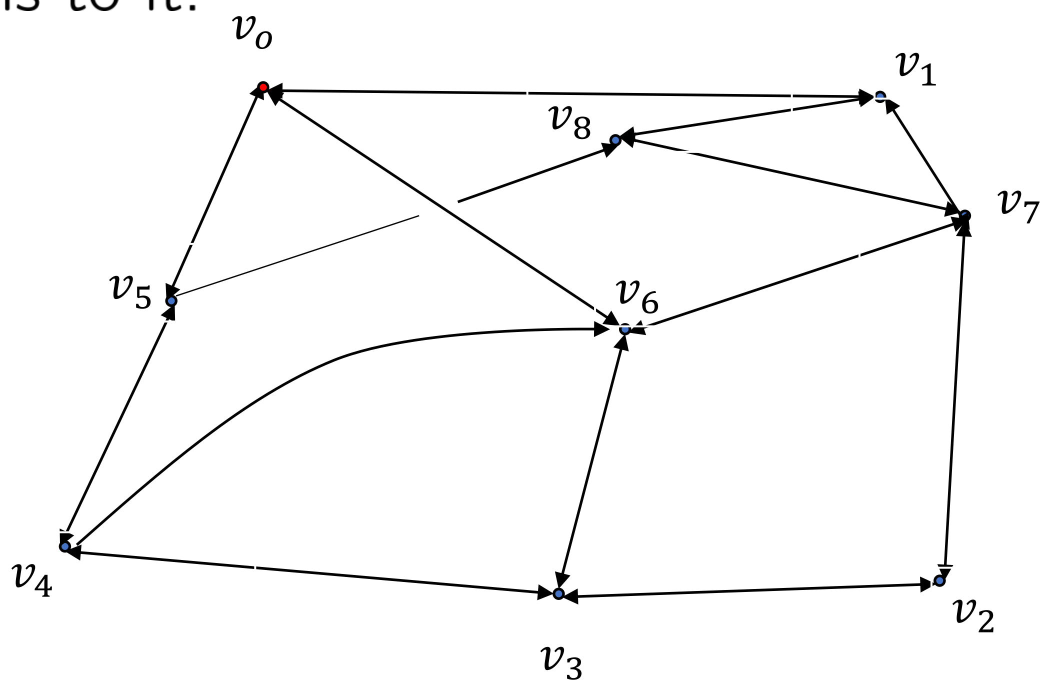
$$g_1(x_1,\dots,x_d) = \sum_{i=1}^{d} \frac{w_i}{[kg]} x_i - MAXWEIGHT \leq 0$$

$$g_2(x_1,\dots,x_d) = \sum_{i=1}^{d} x_i \leq MAXN$$

$$x_i \in \{0,1\}, i = 1,\dots,d$$

# Example: Traveling Salesperson Problem

Given a distance matrix, find a tour of a vehicle that visits every city exactly once and starts from a depot and returns to it.




This can be easily described, but how to model this in terms of discrete (binary) variables and (constraint) functions that depend on them?

# Encoding of a TSP tour

| City 1 | City 2 | City 3 | City 4 |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |

This matrix encodes the tour

$$0 \to 1 \to 2 \to 4 \to 3 \to 0$$

$$\sum_{i=1}^{n-1}\sum_{j=1}^{n}\sum_{k=1}^{n} d_{j,k}x_{i,j}x_{i+1,k} + \sum_{i=1}^{n} x_{1,i}d_{0,i} + \sum_{i=1}^{n} x_{n,i}d_{i,0} \to \min$$

subject to

$$\forall i \in 1,\ldots,n : \sum_{j=1}^{n} x_{i,j} = 1$$

$$\forall j \in 1,\ldots,n : \sum_{i=1}^{n} x_{i,j} = 1$$

$$x_{i,j} \in \{0,1\}^{n\times n}$$

Recall. Previous slide

| City 1 | City 2 | City 3 | City 4 |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 0 |

This matrix encodes the tour

$$0 \to 1 \to 2 \to 4 \to 3 \to 0$$

Here $d_{i,j}$ denotes the distance from $i$ to $j$.and 0 is the index of the depot. If $x_{i,j} = 1$ and $x_{i+1,k} = 1$ then the vehicle drives in the $i$-th step from $j$ to $k$.

Exercise: How does this equation change if one is starting at node 0 and the tour needs to end in other depot, say node $n$?

# Mathematical Programming Models: Exercise

MIN
MAX
Trick

Question: How to solve problems of the following kind  with an LP solver:

$\max\{u_1(x_1, \ldots, x_n), \ldots , u_m(x_1, \ldots, x_n)\} \rightarrow \min$

where $u_1 \ \ldots \ u_m$ are linear functions of the type
$u\_i(\ldots) = a_{i1}\, x_1 + \ \ldots \ a_{im}\, x_i$?

Answer: ….

QP
Trick

Let us assume the problem $a_1\, x_1 \ \ldots \ a_m\, x_m \rightarrow \min$
s.t. $w_1\, x_1 + \ \ldots \ w_n\, x_n \leq c,\ x_i \in \{0,1\}$, i $= 1 \ \ldots \ n$

How can we formulate it as a nonlinear problem with continuous variables?

Answer: ….

# Mathematical Programming Models

MIN
MAX
Trick

Question: How to solve problems of the following kind  with an LP solver:

$\max\{u_1(x_1, \dots, x_n), \dots , u_m(x_1, \dots, x_n)\} \rightarrow \min$

where $u_1$ … $u_m$ are linear functions of the type
$u\_i(\dots) = a_{i1}\, x_1 + \dots\ a_{im}\, x_i$?

$$f(x_1, \dots, x_n, x_{n+1}) = x_{n+1} \rightarrow \min$$

Answer: ….

$$s.t. a_{i1}\mathrm{x}_1 + \cdots \mathrm{a}_{im} x_i \le x_{n+1}, i = 1, \dots, m$$

QP
Trick

Let us assume the problem $a_1\, x_1$ … $a_m\, x_m \rightarrow \min$
s.t. $w_1\, x_1 + \dots\ w_n\, x_n \le c,\ x_i \in \{0,1\}$, $\mathrm{i} = 1 \dots n$

How can we formulate it as a continuous problem?

Answer: ….

$$(1 - x_i)x_i = 0, i = 1, \dots, n$$
$$x_i \in IR$$

# Placement of unit discs on the square – mixed integer problem – 'Big M trick'

How to formulate the problem of placing as many unit discs of radius 1.5m possible on a square of side length 100meter? The unit discs may not overlap.

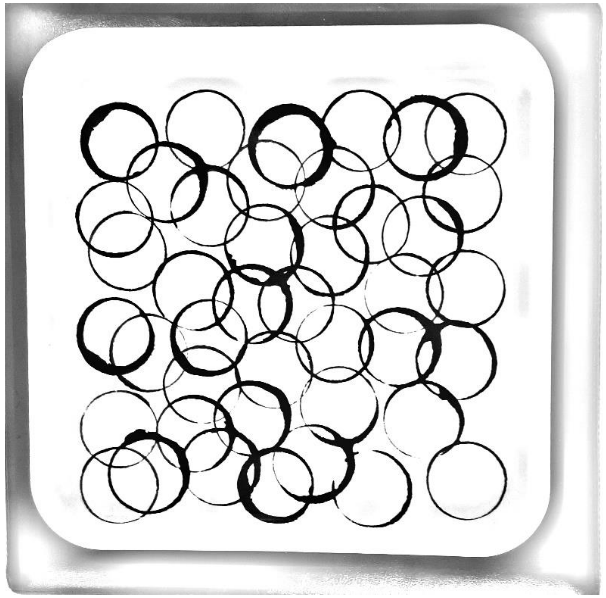
Answer: $Choose\ n = (100 \times 100)/\,(2\pi\ 10^2)$

$$f(b_1, \dots, b_n, x_1, \dots, x_n, y_1, \dots, y_n) = \sum b_i \to max$$

$$g_1(\boldsymbol{b}, \boldsymbol{x}) = b_i b_j \sqrt{\{(x_i - x_j)^2 + (y_i - y_j)^2\}} \leq 1.5$$

$$i = 1, \dots, n-1, j = i+1, \dots, n$$
$$b_i \in \{0,1\}$$
$$x_i \in [10,90]$$
$$y_i \in [10,90]$$

- $n$ is upper bound for number of discs that fit.
- Only if $b_i = 1$ and $b_j = 1$, the distance constraint is checked, otherwise it is always satified (inactive discs cannot overlap)
- But: Highly non-linear!



Unit disc problem: Infeasible design due to overlap.

Alternative QP formulation (big M trick):

$$g_1'^{(\boldsymbol{b},\boldsymbol{x})} = (x_i - x_j)^2 + (y_i - y_j)^2 \leq 2.25 + M(1 - b_i) + M(1 - b_j)$$

Two tricks are used here:

*(1) Big M: choose an M so big that whenever $x_i = 0$ or $x_j = 1$ then $g_1$' is satisfied. What is the minimal value for M?*

*(2) Omit the square root by taking the square on both sides of $g_1$ equation. Values are always positive.*

# Exercise: Linear Programming

**Making Mugs and Bowls**

Mr. Asep makes mugs and bowls. It takes 4 minutes to make a mug and 3 minutes to make a bowl. A mug consists of 1kg of clay and a bowl requires 1.5kg. of clay. He has 20 hours available for making the mugs and bowls and has 450 kg. of clay on hand. He makes a profit of $2 on each mug and $3 on each bowl.

How many mugs and how many bowls should he make in order to maximize her profit? Solve graphically and/or with LP Solve.

x = number of produced mugs
y = number of produced bowls

This homework is only for exercise and not graded. The solution will be discussed in the next lecture.

https://online-optimizer.appspot.com/?model=builtin:default.mod

# Timetabling example (pyomo)

https://towardsdatascience.com/schedule-optimisation-using-linear-programming-in-python-9b3e1bc241e1

**Decision variables**

$$x_{c,s} = \begin{cases} 1 & \text{if case } c \text{ is assigned to session } s \\ 0 & \text{otherwise} \end{cases}$$

$t_c$ = start time of case $c$

$u_s$ = utilisation of session $s$

**Model Parameters**

$C = \{c_1, \dots, c_n\}$ set of all cases assigned to consultant

$S = \{s_1, \dots, s_m\}$ set of all upcoming theatre sessions assigned

$ST_s$ = start time of session $s$

$\Delta T_{C,c}$ = duration of case $c$

$\Delta T_{S,s}$ = duration of session $s$

$D_{C,c}$ = deadline of case $c$

$D_{S,s}$ = date of session $s$

$K_{max}$ = maximum theatre session utilisation (e.g. 0.85)

**Decision variables**

$$x_{c,s} = \begin{cases} 1 & \text{if case } c \text{ is assigned to session } s \\ 0 & \text{otherwise} \end{cases}$$

$t_c$ = start time of case $c$

$u_s$ = utilisation of session $s$

**Objective Function**

$$\text{maximise} \sum_{s=1}^{m} u_s$$

**Constraints**

If $x_{c,s} = 1$ then $\quad t_c \geq ST_s$

If $x_{c,s} = 1$ then $\quad t_c + \Delta T_{C,c} \leq ST_s + \Delta T_{S,s}$

$$\sum_{s=1}^{m} x_{c,s} \leq 1 \quad \forall\, c \in C$$

If $x_{c,s} = 1$ then $\quad D_{S_s} \leq D_{C_c}$

If $x_{c_1,t_1} + x_{c_2,t_2} = 2$ then $\quad [t_{c_1} + \Delta T_{C,c_1} \leq t_{c_2}] \vee [t_{c_2} + \Delta T_{C,c_2} \leq t_{c_1}]$

$$u_s = \frac{1}{\Delta T_{S,s}} \sum_{c=1}^{n} x_{c,s} \Delta T_{C,c}$$

$x_{c,s} \in \{0,1\}, \quad 0 \leq u_s \leq K_{max}, \quad 0 \leq t_c \leq 1440$