

Příklad 1.1 (Výrobní plánování). Podnik vyrábí 2 druhy výrobků V_1, V_2 . Při výrobě se spotřebovávají suroviny S_1 a S_2 a strojový čas Z . Na výrobu 1 kg výrobku V_1 se spotřebují 2 kg suroviny S_1 a 6 kg suroviny S_2 , žádný strojový čas. Doba výroby V_2 je 2 hodiny a spotřebuje se 5 kg suroviny S_1 a 1 kg suroviny S_2 . Na 1 den máme k dispozici 20 kg S_1 , 15 kg S_2 a 7 hodin na zařízení Z . Při prodeji pak získá podnik 2 Kč za 1 kg V_1 a 4 Kč za 1 kg V_2 . Stanovte optimální výrobní plán (tj. stanovte, kolik kg kterého výrobku se má vyrobit, aby byl dosažený zisk maximální).

Řešení. Proměnné modelu x_1, x_2 vyjadřují počet kg výrobků V_1, V_2 , který se bude vyrábět.

```
# our first optimization model:
model = gu.Model("our first model")
x1 = model.addVar(vtype=gu.GRB.CONTINUOUS, lb=0)
x2 = model.addVar(vtype=gu.GRB.CONTINUOUS, lb=0)

model.update()

model.addConstr(2*x1 + 5*x2 <= 20)
model.addConstr(6*x1 + x2 <= 15)
model.addConstr(x2 <= 3.5)

model.setObjective(2*x1 + 4*x2)
model.modelSense = gu.GRB.MAXIMIZE

model.setParam("OutputFlag", 0.0)
model.optimize()
```

naivní implementace
(nehodná pro velké úlohy)

$$\begin{aligned} \max \quad & 2x_1 + 4x_2 && \text{účelová funkce (objective function)} \\ \text{s.t.} \quad & 2x_1 + 5x_2 \leq 20 \\ & 6x_1 + x_2 \leq 15 \\ & 0 \leq x_2 \leq 3.5 \\ & x_1 \geq 0, x_2 \geq 0. \end{aligned}$$

omezení (constraints)
→ množina přípustných řešení
(feasible solutions)

$$C = \begin{pmatrix} 2 \\ 4 \end{pmatrix} \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$A = \begin{pmatrix} 2 & 5 \\ 6 & 1 \\ 0 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 20 \\ 15 \\ 3.5 \end{pmatrix}$$

implementace:

$$\begin{aligned} \max_x \quad & \sum_j c_j x_j \\ \text{s.t.} \quad & \sum_j A_{ij} x_j \leq b_i \quad \forall i \\ & x_j \geq 0 \quad \forall j \end{aligned}$$

maticový (vektorový) zápis:

$$c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$(c_1 \ c_2) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$\begin{aligned} \max \quad & C^T x \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

```
material_ij = np.array([[2, 5], [6, 1], [0, 1]])
|
m, n = np.shape(material_ij)
irange = range(m)
jrange = range(n)

profit_j = [2, 4]
resources_i = [20, 15, 3.5]

model = gu.Model("our first model")
xj = [model.addVar(vtype=gu.GRB.CONTINUOUS, lb=0) for j in jrange]
model.update()

for i in irange:
    model.addConstr(gu.quicksum(material_ij[i][j]*xj[j] for j in jrange) <= resources_i[i])

model.setObjective(gu.quicksum(profit_j[j]*xj[j] for j in jrange))
model.modelSense = gu.GRB.MAXIMIZE

model.setParam("OutputFlag", 0.0)
model.optimize()
```

standardní tvar: ($C \neq c$)

$$\begin{aligned} \min_x \quad & C^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$