Lineární programování I

Pavla Pecherková, Šárka Jozová, Ivan Nagy

Fakulta dopravní

ČVUT, Praha

Obsah

Kapitola 1

Lineární programování

1.1 Formulace úlohy

Nalezněte hodnoty vektoru x, které

1. maximalizují lineární kritérium
$$c'x = c_1x_1 + c_2x_2 + \cdots + c_nx_n,$$
2. splňují vedlejší podmínky $Ax \leq b$, tedy
$$a_{i,1}x_1 + a_{i,2}x_2 + \cdots + a_{i,n}x_n \leq b_i,$$
pro $i = 1, 2, \cdots, m$ (m podmínek pro n proměnných) a
3. jsou nezáporné.

Zápis standardní úlohy je následující:

$$c'x \to \text{opt}$$ [1]

$$Ax \leq b$$ [2]

$$x \geq 0$$ [3]

Poznámka

Omezení vytváří simplex, na kterém se hledá optimum. Lineární kritérium tvoří nadrovinu v prostoru J, kde J je kritérium.. Extrém je ve vrcholu nebo na hranici nebo uteče do nekonečna.

[1] hledáme maximum nebo minimum
[2] nepřekročíme limitující omezení (například nemůžeme vyrobit víc výrobků než na které máme materiál)
[3] hodnota je nezáporná

1.1.1 Příklad

Typický příklad úlohy lineárního programování je následující:

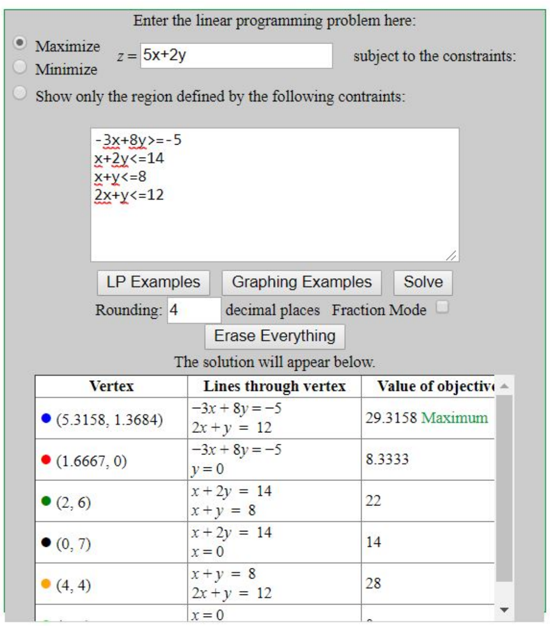
– lineární kritérium

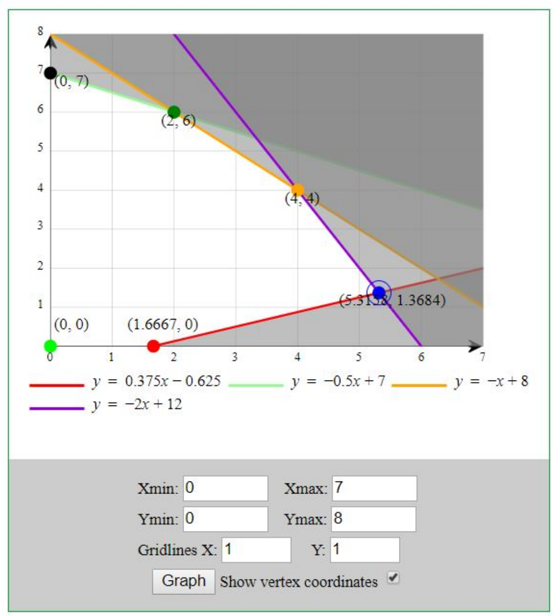
$$5x_1 + 2x_2 \to \max$$

– lineární omezující podmínky

$$\begin{aligned} -3x_1 + 8x_2 &\geq -5 \\ x_1 + 2x_2 &\leq 14 \\ x_1 + x_2 &\leq 8 \\ 2x_1 + x_2 &\leq 12 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Obrázek v https://www.zweigmedia.com/utilities/lpg/index.html?lang=en





Poznámka

První podmínka lze zapsat jako $3x_1 - 2x_2 \leq -4$. Podobný trik lze použít i v kriteriu - lze psát $-(5x_1 + 2x_2) \to \min$.

1.2 Algebraický model

1. *Definice neznámých (rozhodovacích) veličin x*
   V prvé řadě musíme definovat veličiny x, které budeme optimalizovat. Ty mohou být:
   (a) spojité $x \geq 0$ - např. počet ks vyráběného produktu (lze vyrobit i část produktu),
   (b) diskrétní $x \geq 0$, $x \in \mathbb{N}$ - např. počet zařízení s rychlým občerstvením,
   (c) binární $x \in \{0, 1\}$, které nazýváme rozhodovací (0 ne, 1 ano) - např. označení investic z určité množiny, které budou realizovány. Ty budou předmětem dalšího kurzu.
2. *Definice kritéria*
   Kritérium většinou vyjadřuje něco, co chceme maximalizovat (zisk) nebo minimalizovat (náklady, ztráty atd.). Je vyjádřeno jako funkce optimalizovaných veličin x, většinou ve formě $c'x = \sum_i c_i x_i$ jako součet součinů prvků vektorů nebo $\sum_{i,j} c_{ij} x_{ij}$ součet součinů prvků matic. Proměnné c se často nazývají ceny.
3. *Definice omezujících podmínek*
   Vyjadřují podmínky, za kterých má optimalizace probíhat. Nejběžnější jsou
   (a) nezápornost $x \geq 0$ nebo nezápornost celočíselných neznámých $x \in \mathbb{N}$,
   (b) omezení ve tvaru nerovností $Ax \leq b$, kde matice A má rozměry "počet omezení" $\times$ "počet neznámých"
   (c) a případně další, se kterými se postupně setkáme v textu.

1.3 Simplexová metoda

Pro řešení základní úlohy lze využít **simplexovou metodu**, která prochází vrcholy simplexu omezení (v každém kroku pozmění řešení) tak, aby stále narůstala hodnota kritéria. V případě, že ani po změně v dalším kroku nedojde k zvýšení hodnoty kritéria, nalezené řešení uvažujeme jako optimální.

Simplexová metoda může být použita za předpokladu, že jsou splněny základní podmínky:

1. maximalizujeme kritérium (minimalizační úloha lze převést na maximalizační pomocí vzorce $\min f(x) = -\max(-f(x))$,
2. pravé strany musí být nezáporné,
3. omezující podmínky jsou vymezeny nerovnostmi typu $\leq$,
4. proměnné x jsou nezáporné.

Při řešení simplexové úlohy se postupuje v několika krocích. Nejdříve si ověříme, zda platí základní podmínky (viz výše). Poté sestavíme simplexovou tabulku a následuje vlastní řešení úlohy, přičemž úloha se řeší různě podle toho, zda se jedná o standardní nebo nestandardní úlohu, tedy úlohu, kde není splněna podmínka o nezápornosti pravé strany nebo typu nerovnosti. Postup je ukázán pro standardní úlohu. Nestandardní úloha se řeší pomocí dvoufázové simplexové metody. Ta je komplikovanější a zde ji nebudeme podrobně ukazovat.

Poznámka

Simplexová metoda je stále základem řešení úloh lineárního programování. Nicméně její ruční výpočet dnes již nikdo neprovádí, protože pro praktické řešení existuje řada programů. Zde si ji ukážeme spíše z historických důvodů.

1.3.1 Příklad

Řešení standardní simplexové úlohy.

Zadání úlohy:

Určete hodnoty proměnné $x = [x_1, x_2]$, kde $x_1 \geq 0,\ x_2 \geq 0$, která maximalizuje kritérium J

$$J = 2x_1 + 3x_2$$

a vyhovuje podmínkám

$$\begin{aligned} x_1 + x_2 &\leq 5 \\ x_1 + 2x_2 &\leq 8 \end{aligned}$$

Ověření platnosti základních podmínek:

1. maximalizujeme kritérium - ✓ (ano, $2x_1 + 3x_2 \rightarrow \max$),
2. pravé strany musí být nezáporné - ✓ (ano, číslo 5 i 8 jsou kladná čísla),
3. omezující podmínky jsou vymezeny nerovnostmi typu $\leq$ - ✓(ano, v obou rovnicích podmínky jsou uvedeny znaménka $\leq$),
4. proměnné x jsou nezáporné - ✓ (v zadání je definováno, že $x_1 \geq 0,\ x_2 \geq 0$).

Zbavíme se nerovnosti a maxima:

1. Nerovnosti se zbavíme tak, že k levé straně nerovnice přidáme další člen ($s_i > 0$), který bude vždy kladný, v našem případě bude rovnice vypadat takto:

$$\begin{aligned} x_1 + x_2 + s_1 &= 5 \\ x_1 + 2x_2 + s_2 &= 8 \end{aligned}$$

2. Do simplexové tabulky zapíšeme také kritérium. Pokud bychom použili základní tvar, tedy v tomto případě $2x_1 + 3x_2 \rightarrow \max$ bylo by nutné do simplexové tabulky na pravou stranu dát ∞. Z tohoto důvodu se upraví tento řádek rovnice do tvaru[4]

$$-2x_1 - 3x_2 = 0$$

[4] Maximum $2x_1 + 3x_2$ pro $x_1, x_2 \geq 0$ je ∞. Maximum pro $-2x_1 - 3x_2$ pro $x_1, x_2 \geq 0$ je 0.

Sestavení simplexové tabulky:

x_1	x_2	s_1	s_2	RHS	
-2	-3	0	0	0	←řádek kritéria
1	1	1	0	5	←1. omezení
1	2	0	1	8	←2. omezení

Úprava simplexové tabulky:

1. Najdeme nejmenší záporný koeficient v řádku kritéria →*klíčový sloupec.*

   Nejmenší záporný koeficient je koeficient na kterém hodnota kriteria závisí nejvíce. V tomto případě je to hodnota -3. Tomu odpovídá sloupec proměnné x_2.

2. Pro kladné prvky v klíčovém sloupci najdeme nejmenší podíl pravé strany a prvku v klíčovém sloupci →*klíčový řádek.* V případě, že je více klíčových řádků, lze vybrat libovolný z nich.

   Pro první omezení je to podíl $\frac{5}{1} = 5$, pro druhé omezení je to podíl $\frac{8}{2} = 4$. Menší podíl je v řádku s druhým omezením, a to bude tedy klíčový řádek.

3. Na průsečíku klíčového sloupce a klíčového řádku leží *klíčový prvek.*

   V našem případě je tedy klíčový prvek číslo 2.

4. Klíčový řádek celý vydělíme klíčovým prvkem a dostaneme

x_1	x_2	s_1	s_2	RHS
$\frac{1}{2}$	1	0	$\frac{1}{2}$	4

5. Ostatní řádky tabulky redukujeme klíčovým řádkem, tj. klíčový řádek násobíme takovým číslem, aby po přičtení k danému řádku byla v klíčovém sloupci nula.

   Tedy: (i) první řádek má v klíčovém sloupci -3; klíčový řádek násobíme 3 a přičteme k prvnímu řádku; dostaneme prví řádek nové tabulky (dole); (ii) druhý řádek má v klíčovém sloupci 1, klíčový řádek násobíme -1 a přičteme k druhému řádku, dostaneme druhý řádek v nové tabulce; (iii) třetí řádek je klíčový, v něm zůstává upravený klíčový řádek.

   Tady je nová tabulka

x_1	x_2	s_1	s_2	RHS	
-$\frac{1}{2}$	0	0	$\frac{3}{2}$	12	←řádek kritéria
$\frac{1}{2}$	0	1	-$\frac{1}{2}$	1	←1. omezení
$\frac{1}{2}$	1	0	$\frac{1}{2}$	4	←2. omezení

   V nové simplexové tabulce se hodnota kritéria J zvýšila na hodnotu 12. Přesto není řešení ještě ideální, protože v řádku kritéria se dále objevuje záporný koeficient. Z tohoto důvodu budeme znovu upravovat simplexovou tabulku.

6. V případě, že se v řádku kritérií objevuje záporný koeficient (v nové tabulce je to prvek -$\frac{1}{2}$, opakuje se postup od bodu 1 a to do doby, než jsou všechny prvky v řádku kritéria nezáporné.

   V našem případě, po dalším kroku, dostaneme konečnou tabulku ve tvaru

x_1	x_2	s_1	s_2	RHS	
0	0	1	1	13	←řádek kritéria
1	0	2	-1	2	←1. omezení
0	1	-1	1	3	←2. omezení

Protože všechny prvky v řádku kriteria jsou nezáporné, úloha končí.

Výsledek: Optimálním řešení nalezené simplexovou metodou v tomto případě je pro $x_1 = 2$ a $x_2 = 3$. Hodnota kritéria $J = 13$. Řešení dostaneme, když tabulku vyjádříme opět v rovnicích a z nich vypočteme x_1 a x_2.

1.4 Duální úloha

primární úloha	**duální úloha**
$c'x \to \max$ $Ax \leq b$ $x \geq 0$	$b'z \to \max$ $A'z \geq c$ $z \geq 0$

S každým lineárním problémem je spojen odpovídající duální lineární program. Obě úlohy mají společné vektory cen c a omezení b, které ale v duální úloze vystupují v opačných rolích. Také platí, jestliže omezení mají v původní úloze tvar $\leq$ a kritérium se minimalizuje, pak v duální úloze jsou podmínky $\geq$ a kritérium se maximalizuje. Podmínky mají stejnou matici A ale její prvky pro omezení se v původní úloze berou po řádcích, zatímco v duální úloze po sloupcích. Tedy např. (viz následující příklad). Suroviny se transformují na produkty a ty se prodávají. V duální úloze se prodávají přímo suroviny, ale jejich ceny musí být takové, aby se to vyplatilo - tedy větší než ceny výrobků, které by se vyrobili ze stejného množství surovin.

Konkrétně: v primární úloze se maximalizuje součin cena produktu * množství. V duální úloze máme novou proměnnou cena suroviny a máme omezení na výhodnost akce

$$\text{cena suroviny} \geq \text{cena produktu} * \text{množství}$$

Tím se cena dostane do podmínky.

1.4.1 Výroba

Primární úloha

Podnik vyrábí dva druhy výrobku V1 a V2 . Tabulka udává spotřebu surovin S1 a S2 v kg na výrobu 1 ks výrobku V1 , resp. V2 , i disponibilní množství těchto surovin. Zisk z každého výrobku V1 je 3 Kč a z 1 ks V2 je 2 Kč.

	V1	V2	K dispozici
S1	2	5	1000
S2	4	1	1100
Zisk	3	2	$\to$ max

Optimální výrobní plán podniku je, aby dosáhl maximálního zisku. V tom případě se suroviny transformují na výrobky a ty se (se ziskem) prodávají. Primární úloha je

$$3x_1 + 2x_2 \to \max$$

$$\begin{aligned} 2x_1 + 5x_2 &\leq 1000 \\ 4x_1 + x_2 &\leq 1100 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Duální úloha

Zde uvažujme zcela jinak. Úkolem je sestavit úlohu, kdy nebudeme ze surovin vyrábět výrobky a ty prodávat, ale budeme prodávat přímo suroviny, které má podnik k dispozici. Otázkou, která nás zajímá, je, jaké by musely být ceny surovin (a jakého nejmenšího možného zisku přitom dosáhneme), aby se jejich přímý prodej vyplatil. Označme tedy neznámou cenu jednotkového množství suroviny S1 symbolem u_1 a neznámou cenu jednotkového množství suroviny S2 symbolem u_2 . Prodejní cena celkového množství surovin je potom dána funkcí

$$f = 1000u_1 + 1100u_2$$

Aby se přímý prodej surovin vyplatil (proti transformaci surovin na výrobky), je třeba zajistit, aby se přímým prodejem surovin, potřebných pro výrobu 1 ks výrobku V1, dosáhlo alespoň stejného zisku jako v případě jeho výroby a prodeje, tj. alespoň 3 Kč. Dostáváme podmínku

$$2u_1 + 4u_2 \geq 3$$

Analogickou úvahou vzhledem k surovinám potřebných k výrobě výrobku V2 dospějeme k podmínce

$$5u_1 + u_2 \geq 2.$$

Ceny surovin uvažujeme nezáporné, tj. $u_1 \geq 0$, $u_2 \geq 0$.

Dostáváme tedy model

$$f = 1000u_1 + 1100u_2 \to \min$$

$$\begin{aligned} 2u_1 + 4u_2 &\geq 3 \\ 5u_1 + u_2 &\geq 2 \\ u_1, u_2 &\geq 0 \end{aligned}$$

což je duální úloha.

1.4.2 Stravovací problém

Primární úloha

Jak zajistíme nejlevnější stravu, která dodrží základní požadavky na zdravou výživu? Máme k dispozici n potravin s cenami c_j, $j = 1, 2, \cdots, n$. Dále existuje m výživových faktorů. Zdravá potrava musí obsahovat alespoň b_i jednotek každého faktoru $i = 1, 2, \cdots, m$. Přitom potravina j obsahuje a_{ij} faktoru i.

Zavedeme sloupcové vektory c a b a x, kde x_j je počet jednotek potraviny j. Úloha má tvar

$$c'x \to \min$$

$$Ax \geq b$$

$$x \geq 0$$

Duální úloha

Farmaceutická společnost vyrábějící koncentráty výživových faktorů v tabletkách se snaží dělat reklamu svým produktům. Tvrdí, že faktory je lépe získávat z tabletek než z potravin. Demonstruje to tím, že udává ceny $z_1, z_2, \cdots, z_m$ faktorů tak, aby maximalizovaly příjem, ale zároveň mohly konkurovat potravinám. Konkurence potravinám znamená, že cena jednotky potraviny i připravené z výživového faktoru není větší než cena c_i potraviny na trhu. Vektor $a_i = [a_1 \cdots a_n]_i$ označuje kombinaci výživových faktorů potřebných pro přípravu potraviny i. Matice A s řádky a_i tedy říká, jak se faktory kombinují pro výronu jednotlivých jídel. Cena potraviny i je $z'a_i$, ceny všech potravin jsou $z'A$. Pro konkurenci je třeba, aby $z'A \leq c'$. Tržba je $z'b$, kde b jsou potřebná množství faktorů. Problém je

$$z'b \to \max$$

$$z'A \leq c'$$

$$z \geq 0$$

1.4.3 Dopravní problém

Primární úloha

Výrobce určitého zboží má m skladů s kapacitami $a_1, a_2, \cdots, a_m$. Odtud se má přepravit k n odběratelům, kteří požadují množství $b_1, b_2, \cdots, b_n$. Cena přepravy ze skladu i k uživateli j je c_{ij}. Máme navrhnout množství x_{ij} ze skladu i k odběrateli j tak, abychom splnili požadavky odběratelů a minimalizovali cenu za převoz. Úloha je

$$\sum_{ij} c_{ij} x_{ij} \to \min$$

$$\sum_{j} x_{ij} = a_i, \ \forall i$$

$$\sum_{i} x_{ij} = a_j, \ \forall j$$

$$x_{ij} \geq 0$$

Duální úloha

Objeví se podnikatel, který si myslí, že dokáže přepravovat lépe. Nabídne výrobci, že odkoupí zboží ve skladech a dopraví ho do cílů. Cenu zboží určuje podnikatel. Musí volit ceny: u_i (nákup ve skladu i) a v_j (prodej odběrateli j). Aby mohl konkurovat současnému rozvozu, musí platit $u_i + v_j \leq c_{ij}$. Kromě toho musí být ceny takové, aby podnikateli maximalizovaly výnos. Úloha tedy bude

$$\sum_{i=1}^{m} a_i u_i + \sum_{j=1}^{n} b_j v_j \to \max$$

$$u_i + v_j \leq c_{ij}, \forall i, j$$
$$u, v \geq 0$$

Poznámka

1. *Jak může rozvážet lépe, když to původní je optimální? Je to jiný princip, do kterého je zamontován trh.*
2. *A kam se poděl převoz? Ten je implicitní. Zboží se na místě prodá a pak se v cíli koupí a dodá odběrateli.*
3. *Podmínka: Výrobce původně zaplatil cestu c_{ij} a dostal za nákup od odběratele v_j. Tedy jeho zisk je $v_j - c_{ij}$. Teď dostane od podnikatele u_i a dost. To musí být $\geq$ než $v_j - c_{ij}$, a tedy $u_i \geq v_j - c_{ij}$ a tedy $u_i - v_j \geq -c_{ij}$ a nakonec (protože u_i je s mínusem) $u_i + v_i \leq c_{ij}$ - což je uvedená podmínka.*

1.5 Citlivostní analýza

Analýza omezení (citlivost)

Koeficient citlivosti (stínová cena, shadow price, dual price) příslušného omezení ukazuje, o kolik by zvýšila hodnota účelové funkce, kdyby se zvýšila pravá strana omezení o jednotku (v maximalizační úloze, v minimalizační naopak).

Analýza veličin (stabilita)

Koeficient stability (redukovaná cena, reduced costs) vyjadřuje, o kolik by se musela zvýšit (při maximalizaci, při minimalizaci snížit) cena veličiny, aby se daná veličina stala aktivní proměnnou - tj. aby její změna způsobila změnu hodnoty kriteria.

Poznámka

Redukovaná cena je nenulová jen pro veličinu, která je v optimálním řešení omezena svou vlastní nezáporností.

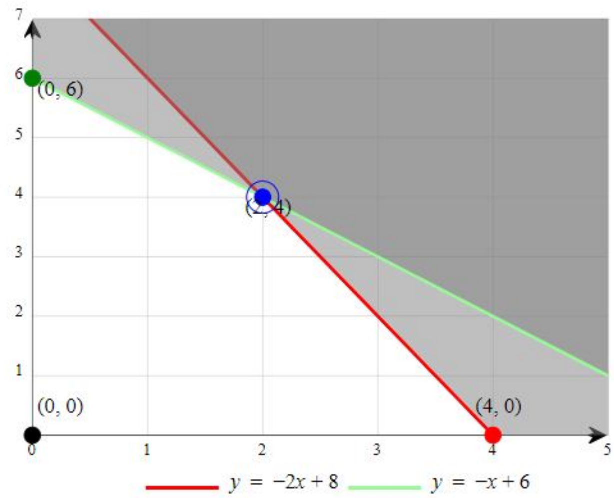
1.5.1 Příklad - citlivost

Uvedené pojmy budeme demonstrovat na příkladě

$$3x + 2y \to \max$$

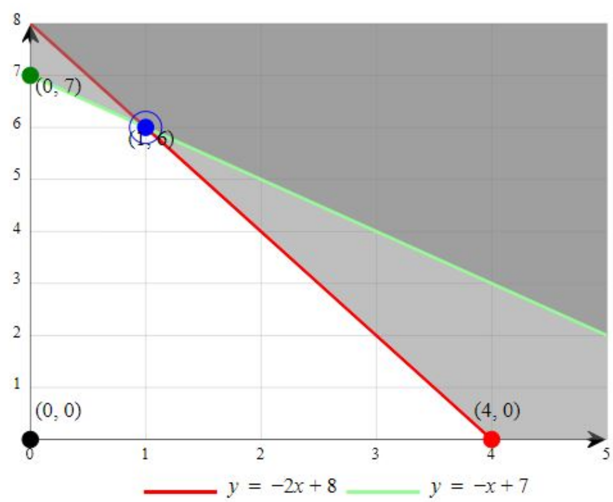
$$2x + y \leq 8$$
$$x + 2y \leq 6$$
$$x \geq 0, y \geq 0$$

Simplex omezení je na obrázku



a podle daného kriteria dostáváme optimální řešení v bodě $(2,4)$ s hodnotou kriteria 14.

Jestliže pravou stranu druhého omezení zvětšíme o jedna (tedy na 7), bude simplex omezení podle následujícího obrázku



Optimální řešení se změní na $(1,6)$, a hodnota kriteria vzroste na 15. Proto **Duální cena** pro druhé omezení bude $15 - 14 = 1$.

Pro první omezení bude také rovna jedné. Ověřte!

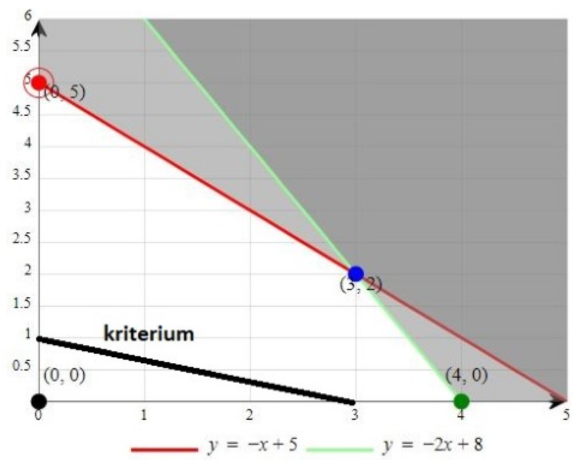
1.5.2 Příklad - stabilita

Uvažujme následující úlohu

$$x + 3y \to \max$$

$$\begin{aligned} x + y &\le 5 \\ 2x + y &\le 8 \end{aligned}$$

Situace je na obrázku



Kritérium roste s vrstevnicí $y = -\frac{1}{3}x$ (tlustá čára) a sice směrem doprava a nahoru. Vrstevnice s nejvyšší hodnotou prochází bodem $(0, 5)$ hranice omezení. Ve směru proměnné x je omezena svou nezáporností.

Optimální řešení $(0, 5)$ dá kritérium $c_1 \cdot 0 + c_2 \cdot 5$. Změna koeficientu c_1 hodnotu kriteria neovlivní. Aby se veličina x_1 stala aktivní, tj. aby změna c_1 začala ovlivňovat hodnotu kriteria, musel by optimální bod řešení přeskočit do dalšího vrcholu simplexu. tedy do bodu $(3, 2)$.

Redukovaná cena veličiny x je dána hodnotou, o kterou by se musel zvýšit odpovídající koeficient cen (tady c_1), aby se veličina x stala aktivní.

V našem případě to evidentně nastane když kriteriální vrstevnice bude rovnoběžná s rovnicí $y = -x + 5$, tj. s rovnicí $x + y = 5$. Tedy kritérium $(1, 3)$ se musí změnit tak, aby druhá souřadnice cen kriteria zůstala stejná (tj. 3) a první souřadnice vytvořila vektor úměrný vektoru $(1, 1)$, což jsou koeficienty zmíněného omezení . Požadovaný vektor tedy bude $(3, 3)$ a jeho první souřadnice se změnila o hodnotu 2. **Redukovaná cena** pro veličinu x tedy bude 2.

Druhá veličina má redukovanou cenu 0. Zdůvodněte!

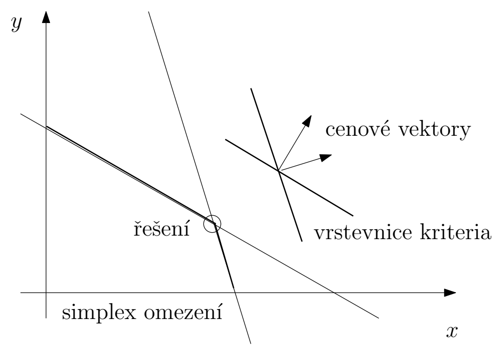
Poznámka

Lze také mluvit o intervalech citlivosti a stability. Jedná se o změny v ***řešení*** *úlohy, když se změní buď pravá strana omezení nebo koeficient cenového vektoru.*

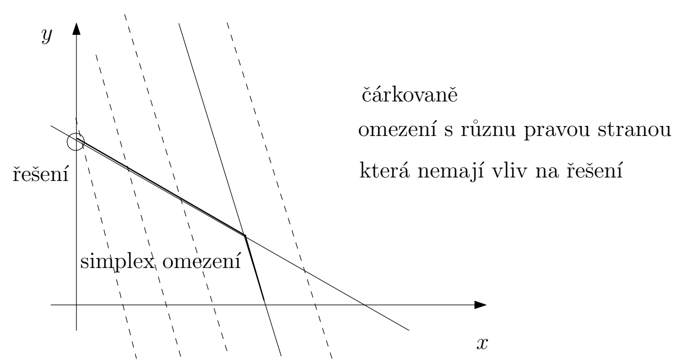
Změna řešení může být dvojí

1. *interval změny ceny ve kterém nedojde ke změně řešení (citlivost),*
2. *interval, ve kterém se může měnit pravá strana, aniž by došlo ke změně řešení (stabilita).*

Případy budeme ilustrovat na obrázcích.
Citlivost



Stabilita



Kapitola 2

Modely lineárního programování

Optimalizační úlohy lineárního programování se liší podle cíle, kterého chceme dosáhnout. Jinak bude vypadat úloha, kde se maximalizuje zisk a jinak úloha, kde se minimalizují náklady. Proto bychom si před začátkem práce měli odpovědět na základní otázky:

1. Co je cílem optimalizace (čeho chceme dosáhnout)?
2. Co jsou řídící veličiny (jak můžeme ovlivnit cíl)?
3. Co nás omezuje (jaké jsou naše limity)?

Odpovědí je několik a dají se rozdělit do několika typů modelů. Právě těmto modelům se bude věnovat následující kapitola, kdy zde budou uvedeny základní optimalizační úlohy. Ke každé úloze bude uvedeno zadání, stručné vysvětlení a připojen odkaz na řešení v jednom z níže uvedených programů.

2.1 Úloha lineárního programování

Uvedeme algebraický model jednoduché lineární úlohy a jeho možná řešení vzhledem k volbě cenových koeficientů. Uvedeme grafické řešení i řešení v tabulkovém procesoru Excel.

2.1.1 Příklad

Obecný příklad demonstrující vliv hodnot ceny c na výsledné řešení.

Určete hodnoty proměnné $x = [x_1, x_2]$, která maximalizuje kritérium

$$J = c_1 x_1 + c_2 x_2 \to \max$$

a vyhovuje podmínkám

$$
\begin{aligned}
2x_1 + x_2 &\leq 8 \\
x_1 + 2x_2 &\leq 6 \\
x_2 &\leq 2
\end{aligned}
$$

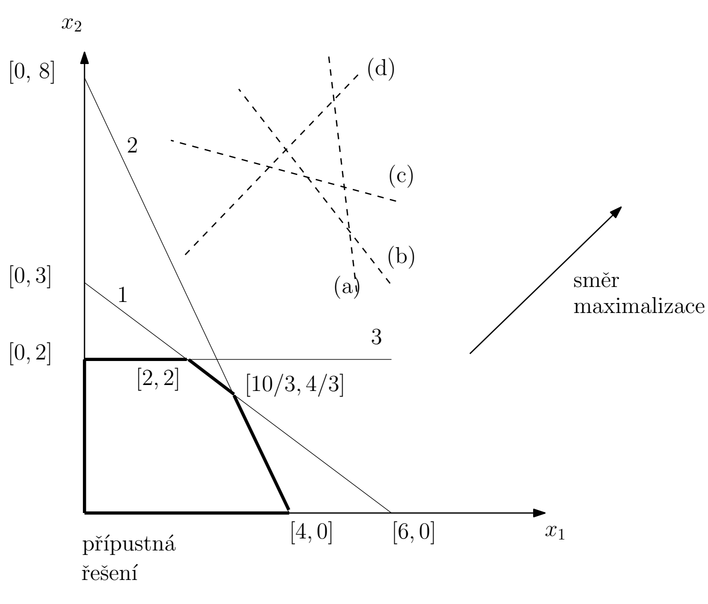
$$x_1 \geq 0,\ x_2 \geq 0$$

Zvolte váhy podle následující varianty a vysvětlete výsledky:

1. $c = [-5,\ 1]$,
2. $c = [1,\ 5]$,
3. $c = [2,\ 3]$,
4. $c = [5,\ 2]$.

Řešení grafické

Geometrická interpretace úlohy je na následujícím obrázku



Řešení varianta 1-4

1. pro simplexovou tabulku (ruční výpočet)

$$\begin{aligned} -5x_1 + x_2 &\to \max \\ 2x_1 + x_2 + s_1 &= 8 \\ x_1 + 2x_2 + s_2 &= 6 \\ x_2 + s_3 &= 2 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

varianta 1

$$\begin{aligned} x_1 + 5x_2 &\to \max \\ 2x_1 + x_2 + s_1 &= 8 \\ x_1 + 2x_2 + s_2 &= 6 \\ x_2 + s_3 &= 2 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

varianta 2

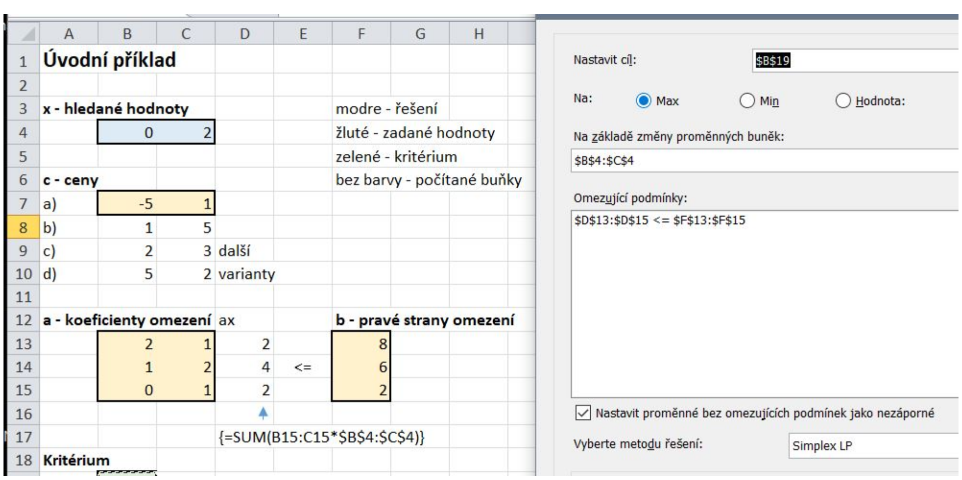
$$\begin{aligned} 2x_1 + 3x_2 &\to \max \\ 2x_1 + x_2 + s_1 &= 8 \\ x_1 + 2x_2 + s_2 &= 6 \\ x_2 + s_3 &= 2 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

varianta 3

$$\begin{aligned} 2x_1 + 3x_2 &\to \max \\ 2x_1 + x_2 + s_1 &= 8 \\ x_1 + 2x_2 + s_2 &= 6 \\ x_2 + s_3 &= 2 \\ x_1 &\geq 0 \\ x_2 &\geq 0 \end{aligned}$$

varianta 4

2. program pro Excel a variantu 1 je (U0a_uvodniPriklad.xlsx)



V programu Excel se mění pouze ceny (žluté pozadí) bez změny podmínek. Hodnota kritéria (zelené pozadí) a hodno ty parametrů x_i se vypočtou po spuštění Řešitele. Výsledek pro variantu 1 je zobrazen nahoře v buňkách s modrým pozadím. Další výsledky sledují postupně rohy omezujícího simplexu

2.2 Optimální produkce

Jedná se o nejzákladnější typ úloh lineárního programování. Vyrábíme určité výrobky, na jejichž konstrukci potřebujeme buď surovinu, kterou platíme nebo stroje. Surovina nebo strojový čas jsou omezeny. Výrobky mají určitou cenu, za kterou je budeme prodávat. Jak vyrábět, abychom dosáhli maximální zisk?

Dále uvedeme různé modifikace této základní úlohy.

2.2.1 Příklad

Optimální produkce dvou výrobků na dvou strojích s omezením strojového času.

V jednom podniku se vyrábí výrobky A a B. K jejich výrobě jsou zapotřebí dva drahé stroje, jejichž využití je omezeno a doba k výrobě jednotlivých výrobků se liší. Jednotlivé časy jsou uvedeny v tabulce

Stroj	Čas strojů potřebný na 1000 výrobků [h]		Čas strojů k využití [h]
	Produkt A	Produkt B	
1	4	6	24
2	4	2	12

Cílem je navrhnout produkci výrobků A a B tak, aby byla *maximální.*

Řešení

Zavedeme proměnnou $x = [x_1, x_2]$ - množství vyráběných výrobků A, B.

Formulace úlohy je potom následující

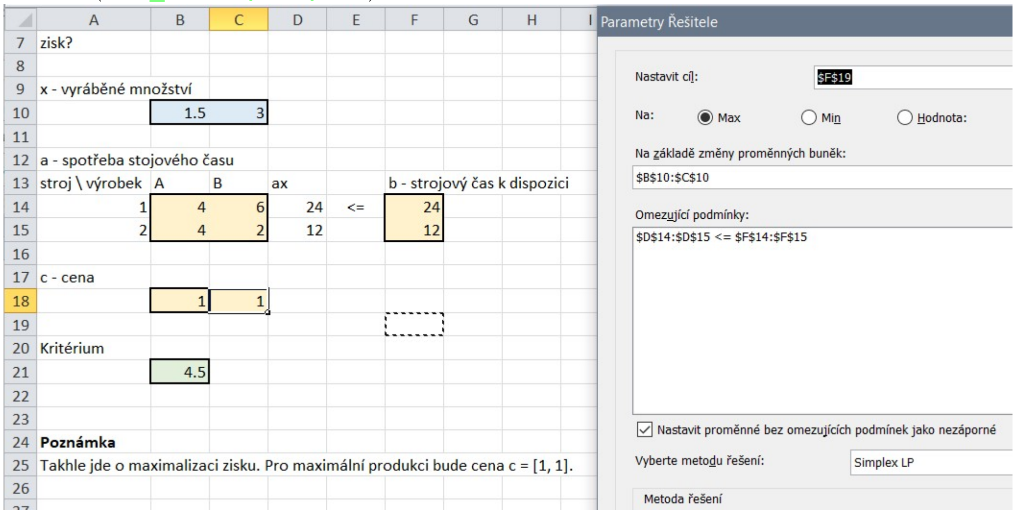
$$x_1 + x_2 \to \max$$

Dále sestavíme podmínky pro lineární programování

$$\begin{aligned} 4x_1 + 6x_2 &\leq 24 \\ 4x_1 + 2x_2 &\leq 12 \end{aligned}$$

$$x_1 \geq 0, \;\; x_2 \geq 0$$

Excel: (U01_prod2Vyrobky.xlsx)



Výsledek je optimální vyráběné množství. Pro výrobek 1 je to 1.5tis. a pro výrobek 2 je 3tis. Optimální hodnota kriteria je 4.5.

2.2.2 Příklad

Rozšíření předchozího příkladu.

Předchozí zadání úlohy je rozšířeno o znalost cen. Ceny výrobků jsou nastaveny následovně: $c_1 = 10$ pro výrobek A a $c_2 = 20$ pro výrobek B. Chceme docílit *maximálního zisku*.

Řešení

Nové kritérium bude

$$10x_1 + 20x_2 \to \max$$

kde x_1 a x_2 je produkce výrobků A a B v tisících a $c_1 = 10$ resp. $c_2 = 20$ je cena výrobku. Podmínky jsou stejné jako v předchozím zadání.

Řešení: $x = [0, 4]$, $J = 80$.

Excel: (U01_prod2Vyrobky.xlsx)

2.2.3 Příklad

Další rozšíření.

Předchozí zadání úlohy (1) je rozšířeno o novou informaci. Podnik podepsal dohodu s dodavatelem, že minimální produkce výrobků bude 1 tisíc (nezapomeňte, že původní produkce byla také v tisících). Náklady na výrobu jednotlivých výrobků jsou $n_1 = 5$ a $n_2 = 10$. Cílem je *minimalizovat výdaje*.

Řešení

Nové kritérium bude

$$5x_1 + 10x_2 \to \min$$

kde x_1 a x_2 je produkce výrobků A a B v tisících a $n_1 = 5$ resp. $n_2 = 10$ je cena výrobku.

$$x_1 \geq 1, \;\; x_2 \geq 1$$

Dále sestavíme podmínky pro lineární programování

$$\begin{aligned} 4x_1 + 6x_2 &\leq 24 \\ 4x_1 + 2x_2 &\leq 12 \end{aligned}$$

$$x_1 \geq 1, \;\; x_2 \geq 1$$

Na rozdíl od původního zadání (1) je zde zakomponováno, že min. produkce je 1 ($x_1 \geq 1$, $x_2 \geq 1$).

Řešení: $x = [1,\, 1]$, $J = 15$

Excel: (U01_prod2Vyrobky.xlsx)

2.2.4 Příklad

Optimální produkce s jedním výrobkem jako zdrojem pro druhý výrobek.

Továrna s dvěma provozy produkuje výrobek A (hliníková trubka), který se prodává samostatně a nebo jako součást pro výrobek B (dopravní značka). Oba výrobky potřebují určitou surovinu S (Al plech), jejíž zdroj je omezen. Kolik suroviny resp. výrobků A je potřeba k výrobě výrobků B je specifikováno v tabulce:

	Spotřeba na jednotku produkce		K dispozici celkem
použitý materiál	výrobek A	výrobek B	
Surovina	5	2	3000
výrobek A	0	1	×
Cena produktu (v tis. Kč)	5	10	×

Vysvětlení tabulky: pro vyrobení výrobku A je potřeba 5 surovin (k výrobě 1 hliníkové trubky je potřeba 5 Al plechů), ale žádný další výrobek A. Pro vyrobení výrobku B je potřeba 2 surovin a 1 výrobek A (k vyrobení celé značky je kromě 2 Al plechů potřeba ještě hliníková trubka). Na skladě je k dispozici 3000 ks surovin (Al plechů). Výrobek A se prodává za 5 000, výrobek B za 10 000.

Dále byla stanovena podmínka, že na konci musíme mít min. 250 ks výrobků A. Počet výrobků B není omezen.

Požadujeme:

1. maximální produkci,
2. maximální zisk.

Podmínky

Stavový vektor bude $x = [x_1, x_2]$ - počty vyrobených výroků A a B (to, co se opravdu vyrobilo - prodávat se bude $x_1 - x_2 + x_2$)

Podmínky jsou společné pro obě varianty, tedy jak pro maximální produkci tak pro maximální zisk

$$5x_1 + 2x_2 \leq 3000$$

$$x_1 - x_2 \geq 250$$

$$x_1 \geq 0, \;\; x_2 \geq 0$$

Řešení 1

Kritérium pro maximální produkci

$$x_1 - x_2 + x_2 = x_1 \to \max,$$

x_1 je počet kusů A a x_2 pro B.

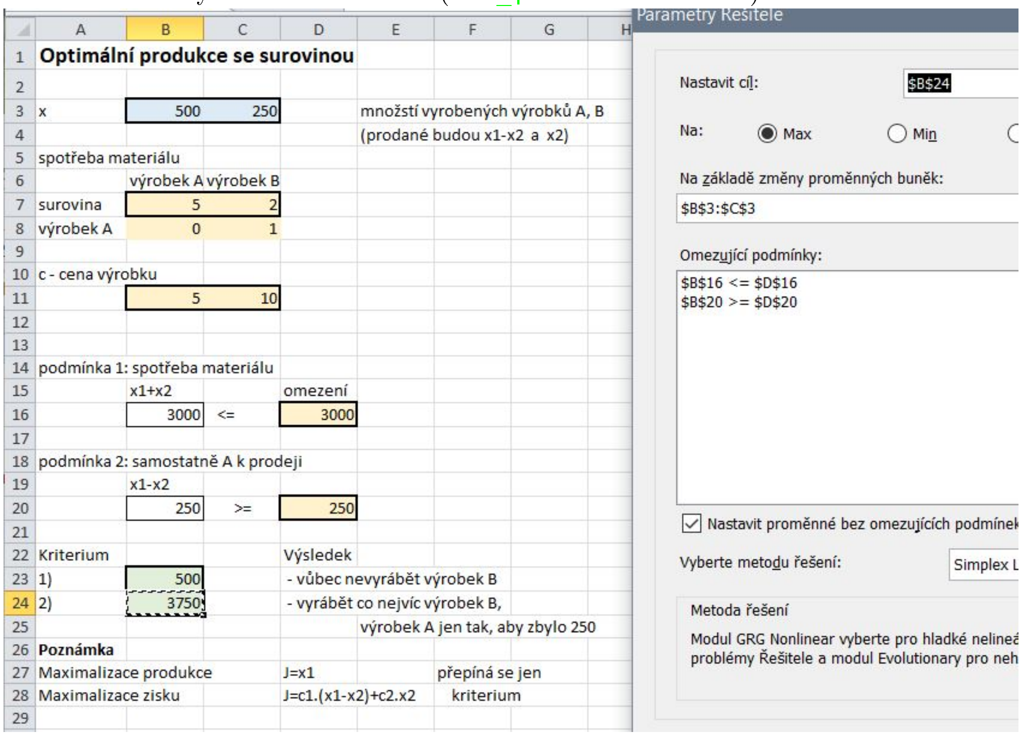
Excel: Podmínky zadáme do Excelu (U02_prodSeSurovinou.xlsx)

Řešení 2

Kritérium pro maximální zisk

$$5(x_1 - x_2) + 10x_2 = 5x_1 + 5x_2 \to \max .$$

Excel: Podmínky zadáme do Excelu (U02_prodSeSurovinou.xlsx)



2.2.5 Příklad

Optimální produkce s extrakcí surovin ze zdrojových látek.

Továrna produkuje dva výrobky B_1 a B_2. Na výrobu B_1 potřebuje látku L_1, na B_2 látku L_2. Tyto látky je možno získat ze surovin A_1, A_2, A_3 a A_4, které je třeba zakoupit. Každá surovina dá jiné množství látek. Specifikace je v tabulce

Látka	Surovina → látka				Potřebné množství
	A_1	A_2	A_3	A_4	látky
L_1 (pro B_1)	2	10	5	20	1000
L_2 (pro B_2)	10	1	1	×	2000
Cena za surovinu	20	10	5	10	

Vysvětlení tabulky: pro vyrobení látky L_1 (resp. L_2) potřebuje 2 (resp. 10) suroviny A_1 nebo 10 (resp. 1) surovin A_2 nebo 5 (resp. 1) surovin A_3 nebo 20 surovin A_4. Víme, že potřebujeme nakoupit takové množství surovin, abych vyrobili min. 1000 látky L_1 resp. 2000 látky L_2. Cena za surovinu je 20 pro A_1, 10 pro A_2, 5 pro A_3 a 10 pro A_4.

Cílem je navrhnout koupi surovin a výrobu tak, abychom minimalizovali výdaje.

Řešení

Stavový vektor $x = [x_1, x_2, x_3, x_4]$ - množství nakoupených surovin A_1, A_2,A_3 a A_4.

Kritérium pro minimální výdaje

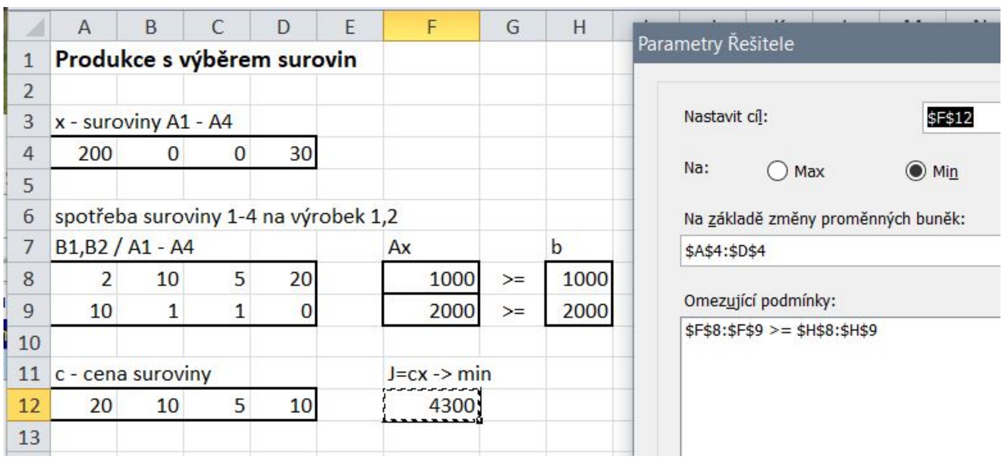
$$20x_1 + 10x_2 + 5x_3 + 10x_4 \to \min$$

kde $x_1, x_2, \ldots$ jsou navrhovaná množství zakoupených surovin.

Dále sestavíme podmínky pro lineární programování

$$\begin{aligned} 2x_1 + 10x_2 + 5x_3 + 20x_4 &\geq 1000 \\ 10x_1 + x_2 + x_3 &\geq 2000 \end{aligned}$$

Excel: (U03_prodVyberS.xlsx)



Řešení: Nakoupíme 200 jednotek suroviny $A1$ a 30 jednotek suroviny $A4$.

2.2.6 Příklad

Výroba tří výrobků
Firma zabývající se strojírenskou výrobou vyrábí (kromě řady dalších) tři výrobky: válce, kotouče a pouzdra, na jejichž výrobě postupně pracují tři druhy strojů: soustruhy, hoblovačky a svářecí stroje. Každý výrobek může být vyráběn několika způsoby, které se liší spotřebou strojového času, nikoliv však cenou výrobku a jeho vlastnostmi. Spotřeba strojového času v hodinách na kus výrobku je zadána v tabulce

	Válce			Kotouče		Pouzdra	
	1	2	3	1	2	1	2
Soustruhy	0.4	0.8	1.2	0.2	–	0.2	–
Hoblovačky	0.4	–	0.2	0.2	0.4	1.2	1.6
Svářecí stoje	0.4	0.6	0.2	1.6	2.0	0.8	0.4

K dispozici je 236 hodin strojového času na soustruzích, 460 na hoblovačkách a 612 na svářecích strojích.

Odbytové ceny jsou 500 korun za jeden válec, 400 korun za jeden kotouč a 600 korun za jedno pouzdro.

Stanovte výrobní program tak, aby firma získala prodejem válců, kotoučů a pouzder maximální tržbu.

Řešení

Zavedeme stavový vektor

$$x = \left[\underbrace{x_1, x_2, x_3}_{válce}, \underbrace{x_4, x_5}_{kotouče}, \underbrace{x_6, x_7}_{pouzdra}\right]$$

jako počet výrobků podle daného programu.

Matici z tabulky (kde prázdná pole doplníme nulami) označíme jako A.

Strojový čas, který je k dispozici, označíme b a odbytové ceny sestavíme do vektoru

$$c = [500, 500, 500, 400, 400, 600, 600]$$

protože cena výrobku je nezávislá na pracovním postupu.

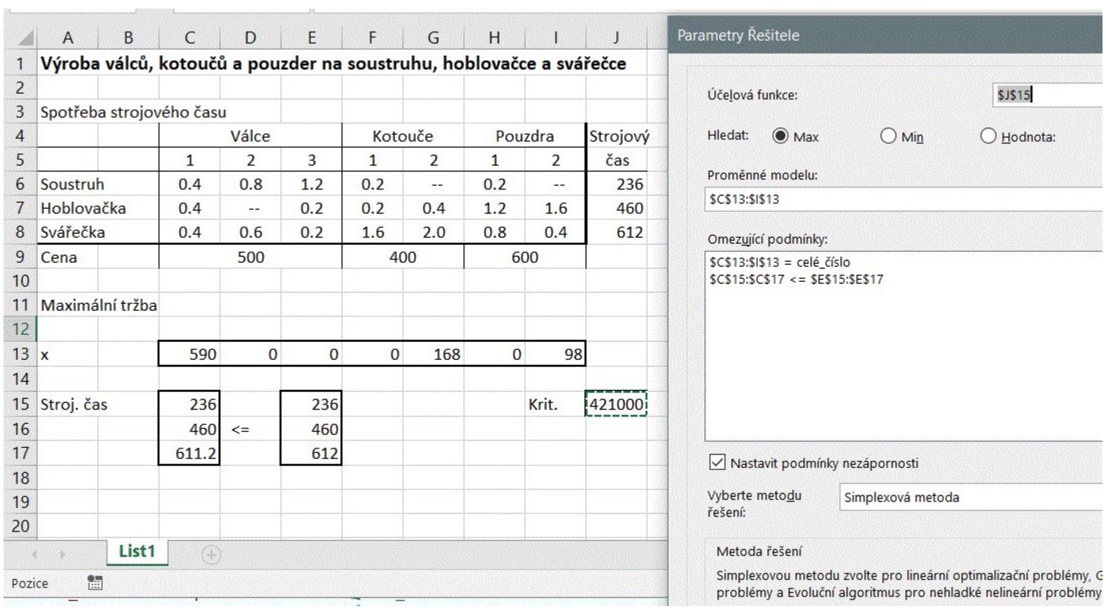
Potom kriterium je

$$J = c'x$$

a podmínky na strojový čas jsou

$$Ax \leq b.$$

Excel: (U04_vyroba3vyrobku.xlsx)



	A	B	C	D	E	F	G	H	I	J
1	**Výroba válců, kotoučů a pouzder na soustruhu, hoblovačce a svářečce**									
2										
3	Spotřeba strojového času									
4				Válce		Kotouče		Pouzdra		Strojový
5			1	2	3	1	2	1	2	čas
6	Soustruh		0.4	0.8	1.2	0.2	--	0.2	--	236
7	Hoblovačka		0.4	--	0.2	0.2	0.4	1.2	1.6	460
8	Svářečka		0.4	0.6	0.2	1.6	2.0	0.8	0.4	612
9	Cena			500		400		600		
10										
11	Maximální tržba									
12										
13	x		590	0	0	0	168	0	98	
14										
15	Stroj. čas		236		236				Krit.	421000
16			460	<=	460					
17			611.2		612					
18										
19										
20										

Pokračování

Firma je vázána smlouvou se zákazníkem, který odebírá výrobky v kompletech. Každý komplet obsahuje 10 válců, 20 kotoučů a 15 pouzder. Kromě zmíněných kompletů, které prodává za cenu 25500 korun, musí firma prodávat i samostatné výrobky jako náhradní díly (tady minimálně 10ks od každého výrobku) za jejich odbytovou cenu. Jaký výrobní program zajistí firmě maximální tržbu?

Zavedeme celkem 3 stavové proměnné:

$x = [x_1, \cdots, x_7]$ jako počet samostatných výrobků podle plánů.

$z = [z_1, \cdots, z_7]$ jako počet výrobků pro komplety.

m - jako počet vyrobených kompletů.

Spotřeba strojů při samostatné výrobě

$$S_1 = \sum_j A_{ij} x_j, \ \forall i$$

Spotřeba strojů při výrobě dílů pro komplety

$$S_2 = \sum_j A_{ij} z_j, \ \forall i$$

Celková spotřeba strojů (omezena strojovým časem)

$$S_1 + S_2 \leq b,$$

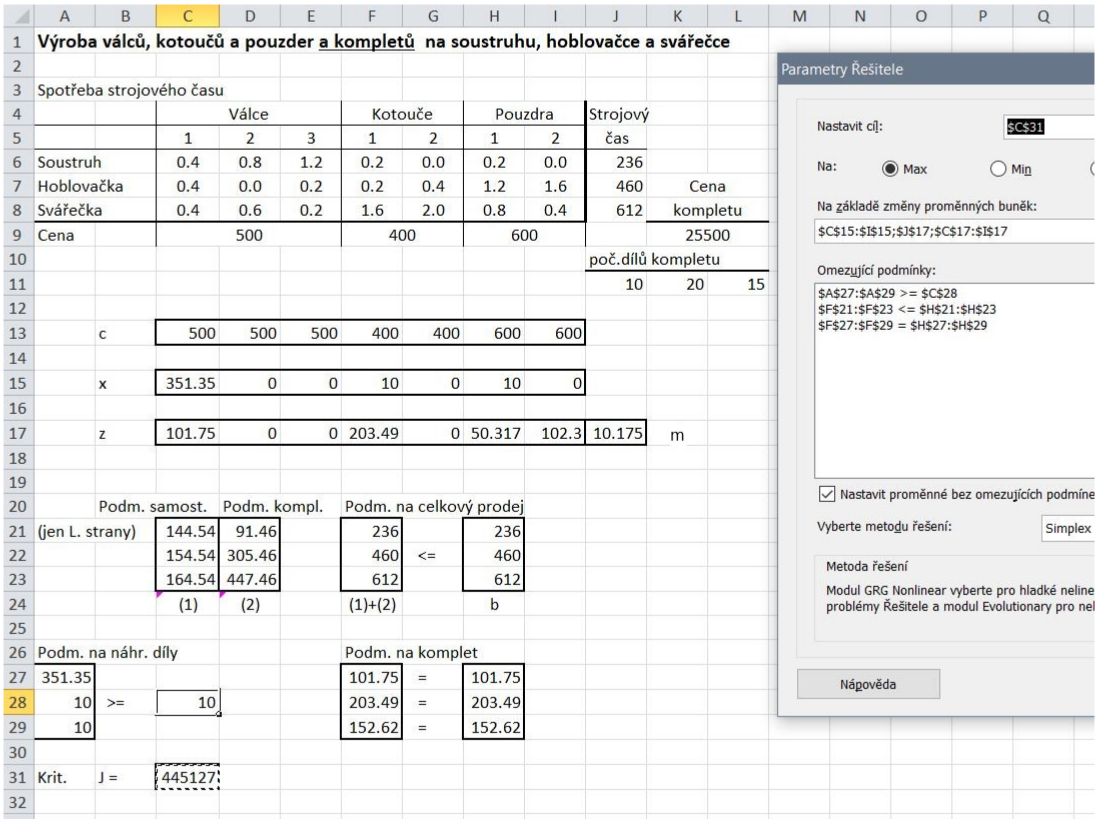
kde $b = [236, 460, 612]'$.

Podmínka na komplety (počty dílů: 10, 20, 15)

$$\begin{aligned} 10\,(z_1 + z_2 + z_3) &= m \\ 20\,(z_4 + z_5) &= m \\ 15\,(z_6 + z_7) &= m \end{aligned}$$

$\cdots$ jednotlivých dílů pro komplety musí být vyrobeno tolik, aby z nich šly vyrobit vždy celé komplety - tedy $10j$ válců, $20j$ kotoučů a $15j$ pouzder. Necelý výsledek v řešení je diskutován v poznámce na konci příkladu.

Excel - řešení včetně kompletů (U05_v3vPokracovani.xlsx)



Poznámka ***(důležitá)***

Řešení x vychází neceločíselné, což úplně neodpovídá našim požadavkům. Nelze vyrobit a nabízet kousek dílu. Celočíselné řešení k problému LP existuje, ale je třeba použít jinou metodu, než jen simplexovou tabulku. Jmenuje se metoda větví a mezí a je dosti složitá. Počítá úlohu LP opakovaně za sebou a při tom vylučuje zlomková řešení. Navíc přináší veliké možnosti při řešení úloh s rozhodováním. Tímto, tzv. Celočíselným programováním se budeme zabývat v dalším kurzu. Zatím nám bude stačit, když řešení zaokrouhlíme. Nemusí to sice být optimální řešení, ale každopádně je mu velice blízko

2.2.7 Příklad

Problém výroby a skladování na dobu tří měsíců.

Firma plánuje na 3 měsíce výrobu dvou výrobků A a B. Jejich produkované množství, jednotková cena a očekávaný odbyt jsou v tabulce

	Měsíc 1		Měsíc 2		Měsíc 3	
	Výr. A	Výr. B	Výr. A	Výr. B	Výr. A	Výr. B
Max. produkce (ks)	70	40	80	30	80	10
Náklady ($/ks)	3.10	10.50	3.20	10.80	3.80	12.00
Odbyt (ks)	50	30	60	10	100	40

Výrobky lze vyrábět do zásoby, ale za uskladnění se na konci měsíce platí. Cena za skladování výrobku A je 20 centů/ks, za B je 50 centů/ks. Na začátku výroby máme k dispozici zásoby 0 ks výrobku A a 10 ks výrobku B. Na konci požadujeme zásoby nulové. Celkové zásoby (A i B dohromady) nesmí přesáhnout 40 ks na konci měsíce 1 a 50 ks na konci měsíce 2. Cílem je vyrábět co nejlevněji.

Řešení

Zavedeme stavový vektor s následující strukturou

$$x = [v_{A1}, v_{A2}, v_{A3}, v_{B1}, v_{B2}, v_{B3}, z_{A2}, z_{A3}, z_{B2}, z_{B3}]\,,$$

kde v je výroba, z jsou zásoby a index značí výrobek a měsíc. Tedy např v_{B1} znamená výroba B v měsíci 1 a z_{A2} jsou zásoby A na **začátku** měsíce 2.

Platí se za náklady na výrobu a skladování. Podle struktury vektoru x seřadíme příslušné cenové koeficienty

$$c = \left[\underbrace{3.1,\ 3.2,\ 3.8,\ 10.5,\ 10.8,\ 12.0}_{náklady\,na\,výrobu},\ \underbrace{0.2,\ 0.2,\ 0.5,\ 0,5}_{skladování}\right]'$$

Potom kriterium je

$$J = c'x \rightarrow \min$$

Omezení se týkají:

1. maximální produkce

$$x_{A1,2,3} \leq [70, 80, 80]$$

$$x_{B1,2,3} \leq [40, 30, 10]$$

2. velikosti zásob

   (a) výrobek A první měsíc:
   na začátku je zásoba 0; první měsíc se vyrobí v_{A1} a prodá se 50. Tedy zásoba na začátku 2. měsíce z_{A2} bude

$$z_{A2} = 0 + v_{A1} - 50.$$

   (b) výrobek A druhý měsíc:
   na začátku 2. měsíce je zásoba z_{A2}; vyrobí se v_{A2} a prodá 60. Tedy bude

$$z_{A3} = z_{A2} + v_{A2} - 60$$

   (c) výrobek A třetí měsíc
   zásoba na začátku je z_{A3}; vyrobí se v_{A3} a prodá se 100. Na konci musí být zásoby 0. Tedy platí

$$0 = z_{A3} + v_{A3} - 100$$

(d) Podobně pro výrobek B

$$z_{B2} = 10 + v_{B1} - 30$$

$$z_{B3} = z_{B2} + v_{B2} - 10$$

$$0 = z_{B3} + v_{B3} - 40$$

3. maximální povolené zásoby

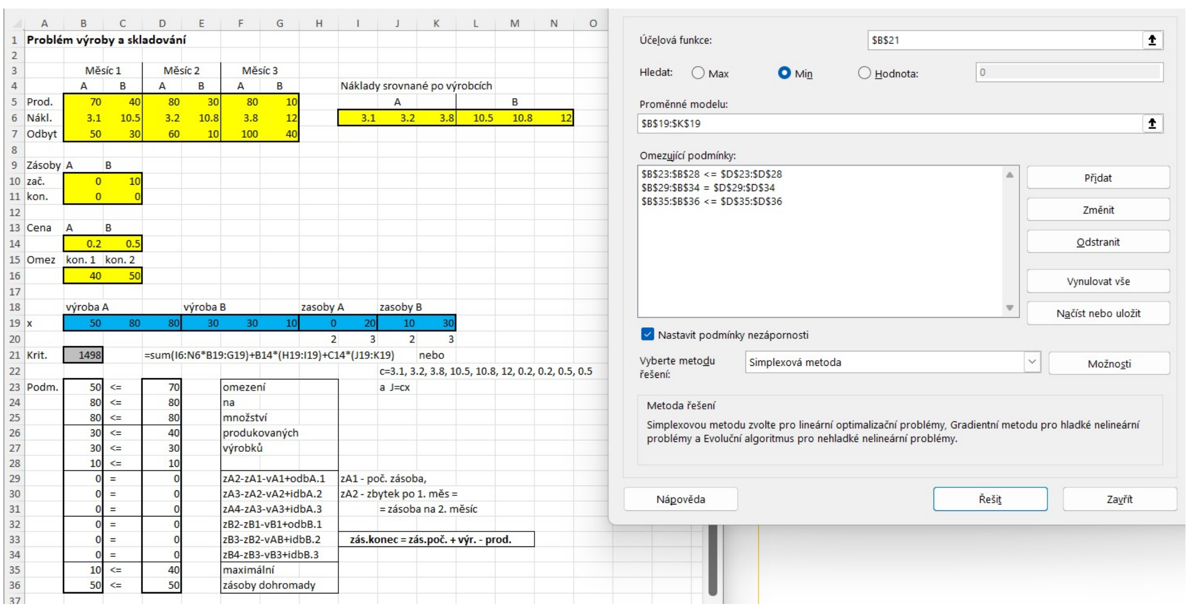
(a) na začátku 2. měsíce

$$z_{A2} + z_{B2} \leq 40$$

(b) na začátku 3. měsíce

$$z_A + z_{B3} \leq 50$$

Excel (U06_vyrobaASklady.xlsx)



2.3 Optimální míchání

Obecným základem tohoto problému je, že máme k dispozici určité suroviny nebo látky obsažení v surovinách a tyto látky mícháme v určitých poměrech tak, abychom získali konečné produkty. Jedná se typicky o míchání krmných směsí, kávy, čaje apod.

Jako optimalizovaný stav zde volíme matici s prvky x_{ij}, kde x_{ij} označuje množství ingredience i, použité v j-tém produktu. Většinou jde o jeden produkt a potom stav x je vektor s prvky odpovídajícími míchaným látkám.

2.3.1 Příklad

Krmná směs.

Pro výkrm jednoho kusu dobytka je zapotřebí 2.5 kg krmné směsi a 240 g proteinů na den. Pro krmení se používá píce a kukuřice. Další specifikace jsou v tabulce

látky \ krmivo	Píce (x_1 kg)	Kukuřice (x_2 kg)	Min. množství (kg)
Krmná směs/kg	1	1.25	2.5
Proteiny/kg	0.4	0.08	0.24
Cena Kč/kg	0.5	0.4	

Vysvětlení tabulky: Z jednoho kg píce se získá 1 kg krmné směsi a 400 g proteinů. Z jednoho kg kukuřice se získá 1.25 kg krmné směsi a 80 g proteinů. Cena jednoho kg píce je 0.50 Kč a kukuřice 0.40 Kč.

Cílem je vytvořit takovou krmnou směs, která bude za splnění požadovaných podmínek nejlevnější.

Řešení

Kritérium pro minimální výdaje

$$0.5x_1 + 0.4x_2 \to \min$$

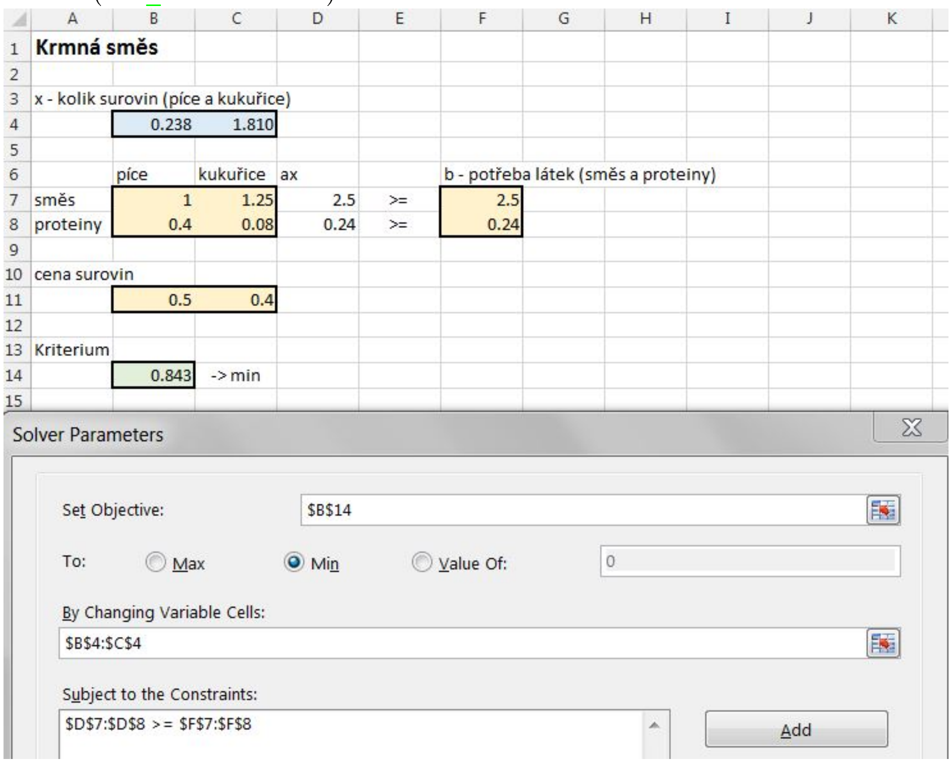
kde x_1 značí množství píce, x_2 množství kukuřice.

Dále sestavíme podmínky pro lineární programování

$$\begin{aligned} x_1 + 1.25x_2 &\geq 2.5 \\ 0.4x_1 + 0.08x_2 &\geq 0.24 \end{aligned}$$

V případě, že nechceme žádné zbytky, budeme mít přesné množství, místo $\geq$ použijeme znaménko $=$.

Excel: (U07_krmnaSmes.xlsx)



Výsledek: Budeme potřebovat 0.238 kg píce a 1.81 kg kukuřice.

2.3.2 Příklad

Optimální strava pro vězně.

Ve vězení navrhují optimální stravu vězňů. Chtějí ji kombinovat z mléka, fazolí a pomerančů. Při tom je třeba dodržet minimální požadavky na obsah stravy podle zákona. Jedná se o vitaminy: B_3, B_1 a C. Obsah těchto vitaminů v jednotlivých složkách stravy je následující

	mléko $[l]$	fazole $[100\,g]$	pomeranče $[1\,ks]$	požadavek
B_3	3.2	4.9	0.8	13
B_1	1.12	1.3	0.19	1.5
C	32	0	93	45
cena (v $)	2	0.2	0.25	

Vysvětlení tabulky: Z jednoho litru mléka se získá 3.2 vitamínu B_3, 1.12 vitamínu B_1 a 32 vitamínu C. Obdobně je to pro 100g fazolí a jeden pomeranč. Cena za sklenici mléka je 2 $, za 100g fazolí je to 0.2 $ a za 1 ks pomeranče je to 0.25 $. Aby byly splněny zákonné požadavky je nutné dodat 13 jednotek vitamínu B_3, 1.5 jednotek vitamínu B_1 a 45 jednotek vitamínu C.

Cílem je namíchat stravu tak, aby její cena byla co nejmenší a zároveň byl splněn zákon?[1]

Řešení

Kritérium pro minimální výdaje

[1]hodnoty vitaminů i cena je pouze orientační a neodráží skutečnost.

$$2x_1 + 0.2x_2 + 0.25x_3 \to \min$$

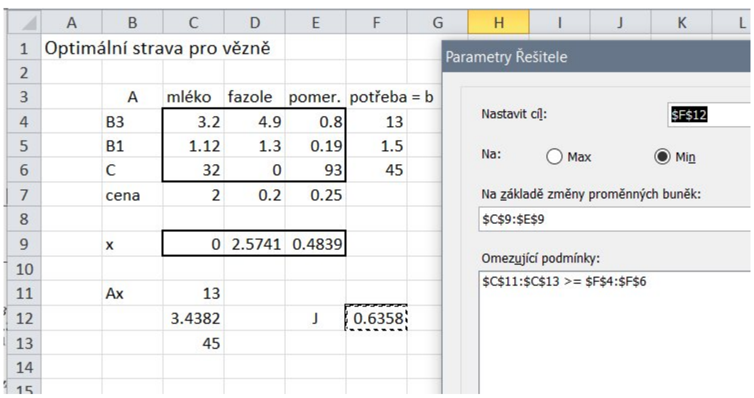
kde x_1 značí množství mléka, x_2 množství fazolí a $x_3$množství pomerančů.

Dále sestavíme podmínky pro lineární programování

$$\begin{aligned}
3.2x_1 + 4.9x_2 + \;\; 0.8x_3 &\geq 13 \\
1.12x_1 + 1.3x_2 + 0.19x_3 &\geq 1.5 \\
32x_1 + \qquad\quad + \;\; 93x_3 &\geq 45
\end{aligned}$$

V případě, že chceme přesné zákonné hodnoty, tak místo $\geq$ použijeme znaménko $=$.

Excel: BUDE (U08_stravaProVezne.xlsx)



Minimální ceny se dosáhne pro 250g fazolí a polovinu pomeranče.

2.3.3 Příklad

Strava pro brigádníky.

Máme navrhnout stravu pro brigádníky a máme k dispozici Hamburgry, Hranolky a Bagety. Jedna porce každého z těchto jídel má určité kalorie a tuky a stanovenou cenu podle tabulky

	Hamburger	Hranolky	Bageta	Požadavky
Kalorie (cal)	250	380	257	(1800, 2200)
Tuky (%)	13	31	28	$\leq$100
Cena ($)	1.59	2.19	2.99	

Cílem je navrhnout co nejlevnější stravu při dodržení požadavků na stravu.

Řešení

Stav úlohy x - kolik porcí hamburgrů, hranolků a baget.

kritérium

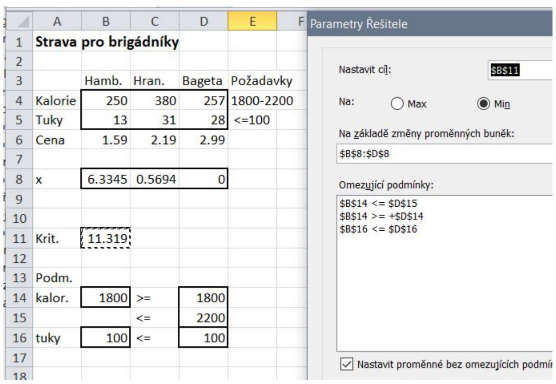
$$1.59x_1 + 2.19x_2 + 2.99x_3 \to \min$$

Podmínky

$$250x_1 + 380x_2 + 257x_3 \geq 1800$$
$$250x_1 + 380x_2 + 257x_3 \leq 2200$$
$$13x_1 + 31x_2 + 28x_3 \leq 100$$

$$x_1 \cdots x_3 \geq 0$$

Excel: (U09_stravaBrigada.xlsx)



Strava bude sestavena z hamburgrů a hranolků v množství 6.33 a 0.57 porcí.

2.3.4 Příklad

Optimální strava v reálu.
Řešíme optimální stravu s hodnotami podle tabulky (v určitých jednotkách)

	Těsto-viny	Rajský protlak	Polévka z ústřic	Hovězí plátek	Mléko	Pom. džus	Jablka	Hranolky
Kalorie	300	60	220	259	110	132	55	152
Tuky (g)	1	0	13	16.3	2.5	0	0.22	9.8
Chol. (mg)	0	0	5	89	10	0	0	0
Sodík (mg)	1	650	790	95	120	5	1.1	168.4
Karbohyd. (g)	63	12	19	20	12	33.4	14.6	15
Vláknina (g)	3	3	2	0	0	0	2.5	1.3
Proteiny (g)	11	2	5	26.1	9	0.5	0.3	2
Vit. A (%)	0	8	2	1	10	2	1	0
Vit. C (%)	0	30	2	0	0	62	8	15
Kalcium (%)	2	2	2	1	30	0	1	1
Železo (%)	20	15	8	17	0	2	1	3
Cena/jednotku	19	56	90	82	51	53	37	32

Řešení

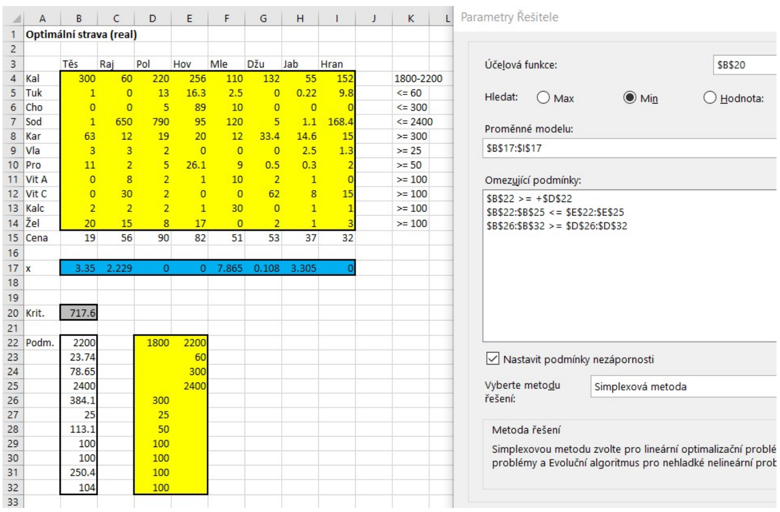
Podmínky řešení jsou

1. Kalorie v intervalu (1800, 2200).

2. Ne více než 65 g tuků.
3. Ne více než 300 mg cholesterolu.
4. Ne více než 2400 mg sodíku.
5. Alespoň 300 g karbohydrátů.
6. Alespoň 25 g vláknin.
7. Alespoň 50 g proteinů.
8. Alespoň 100% doporučené dávky vitamínů A, C, kalcia a železa (každého zvlášť).

Model pro řešení úlohy a jeho implementace v Excelu jsou přímočaré.

Excel: (U10_stravaVRealu.xlsx)



Optimální strava (real)

	Těs	Raj	Pol	Hov	Mle	Džu	Jab	Hran	
Kal	300	60	220	256	110	132	55	152	1800-2200
Tuk	1	0	13	16.3	2.5	0	0.22	9.8	<= 60
Cho	0	0	5	89	10	0	0	0	<= 300
Sod	1	650	790	95	120	5	1.1	168.4	<= 2400
Kar	63	12	19	20	12	33.4	14.6	15	>= 300
Vla	3	3	2	0	0	0	2.5	1.3	>= 25
Pro	11	2	5	26.1	9	0.5	0.3	2	>= 50
Vit A	0	8	2	1	10	2	1	0	>= 100
Vit C	0	30	2	0	0	62	8	15	>= 100
Kalc	2	2	2	1	30	0	1	1	>= 100
Žel	20	15	8	17	0	2	1	3	>= 100
Cena	19	56	90	82	51	53	37	32	
x	3.35	2.229	0	0	7.865	0.108	3.305	0	
Krit.	717.6								
Podm.	2200		1800	2200					
	23.74			60					
	78.65			300					
	2400			2400					
	384.1		300						
	25		25						
	113.1		50						
	100		100						
	100		100						
	250.4		100						
	104		100						

2.3.5 Příklad

Optimální hnojivo pro fotbalové hřiště.

Správce fotbalového stadionu musí zajistit pravidelné hnojení trávníku. Údržba trávníku vyžaduje, aby celková dávka hnojiv byla minimálně 100 kg dusíku, 120 kg fosforu a 150 kg draslíku. Na trhu jsou k dispozici dva druhy hnojiva. První z nich se dodává v pytlích o váze 50 kg a obsahuje po 8 % všech uvažovaných komponent. Cena pytle je 800 Kč. Druhý druh je v pytlích o váze 25 kg a obsahuje 4 % dusíku, 6 % fosforu a 16 % draslíku. Cena jednoho pytle je 320 Kč. Jak nakoupit, aby se pohnojilo a aby cena byla minimální?

Řešení

Stav x zvolíme jako počty pytlů každého z druhů hnojiva

$$x = [x_1, x_2]'$$

Protože některé z dalších výpočtů jsou spojeny s váhou v kg, přepočteme pytle na kilogramy

$$z = [50x_1, 25x_2]'$$

Procentuální obsah živin v hnojivech - jako podíl - je dán v následující tabulce (matici označíme A)

A	dusík	fosfor	draslík
Pytel 1	0.08	0.08	0.08
Pytel 2	0.04	0.06	0.16

Omezení (v kilogramech)

$$b = [100,\ 120,\ 150]'$$

Ceny (v pytlích)

$$c = [800,\ 320]'$$

Úloha LP potom má tvar

– kriterium

$$J{=}c'x \to \min$$

- omezení

$$z'A >= b$$

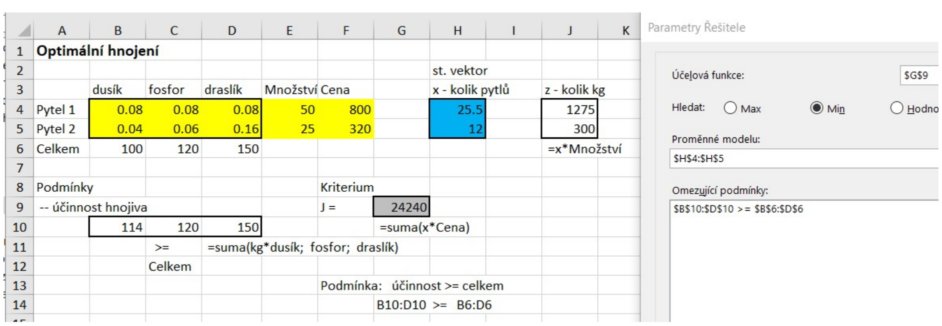
Rozepsáno

$$0.08z_1 + 0.04z_2 \geq 100$$

$$0.08z_1 + 0.06z_2 \geq 120$$

$$0.08z_1 + 0.16z_2 \geq 150$$

Excel (U11_optimalniHnojivo.xlsx)



2.3.6 Příklad

Míchání vína.

Chceme připravit dvě výběrová vína Filzener a Leiwener. Ta vzniknou mícháním vín Riesling, Muller a Silvaner. Údaje o množství a poměrech míchání jsou v tabulce

	Filzener	Leiwener	množství (l)	cena (Kč)
Riesling	(45% − 55%)	(20% − 50%)	10 tis.	80
Muller	(10% − 15%)	(10% − 60%)	5 tis.	60
Silvaner	(35% − 35%)	(30% − 40%)	6 tis.	50
množství (l)	7 tis.	8 tis.		
cena (Kč)	160	180		

Jak vína namíchat tak, abychom dosáhli maximálního ekonomického efektu, tj. abychom na výsledném produktu co nejvíce získali a přitom do zdrojů investovali co nejméně.

Řešení

Stavovou proměnnou zavedeme jako matici x s prvky $x_{i,j}$ s významem: množství zdroje i obsažené v cíli j, tedy např. $x_{2,1}$ označuje množství zdrojového vína Muller v cílovém vínu Filzener.

Dále označíme z (zdroj) celkové množství zdrojových vín

$$z_i = \sum_j x_{i,j},\ i = 1, 2, 3$$

a c (cíl) celkové množství cílových vín

$$c_j = \sum_i x_{i,j},\ j = 1, 2$$

Kriterium J má vyjadřovat maximální zisk, tedy příjem - náklady

$$J = 160c_1 + 180c_2 - 80z_1 - 60z_2 - 50z_3 \to \max$$

Podmínky míchání lze vyjádřit takto:

- obsah zdroje Riesling v cíli Filzener
  $x_{1,1}$ je obsah složky Riesling v cílovém vínu Filzener, kterého je celkem c_1. Tento obsah má být
  - větší, než 45%, a tedy (trojčlenka: $c_1 - 100\%$, $x_{1,1} - 45\%$)

    $$x_{1,1} \geq 0.45c_1$$

  - menší, než 55%

    $$x_{1,1} \leq 0.55c_1$$

- podobně i pro ostatní složky - cíl Filzener

  $$x_{2,1} \geq 0.1c_1,\ x_{2,1} \leq 0.15c_1$$

  $$x_{3,1} = 0.35c_1$$

- cíl Leiwener

  $$x_{1,2} \geq 0.2c_2,\ x_{1,2} \leq 0.5c_2$$

  $$x_{2,2} \geq 0.1c_2,\ x_{2,2} \leq 0.6c_2$$

  $$x_{3,2} \geq 0.3c_2,\ x_{3,2} \leq 0.4c_2$$

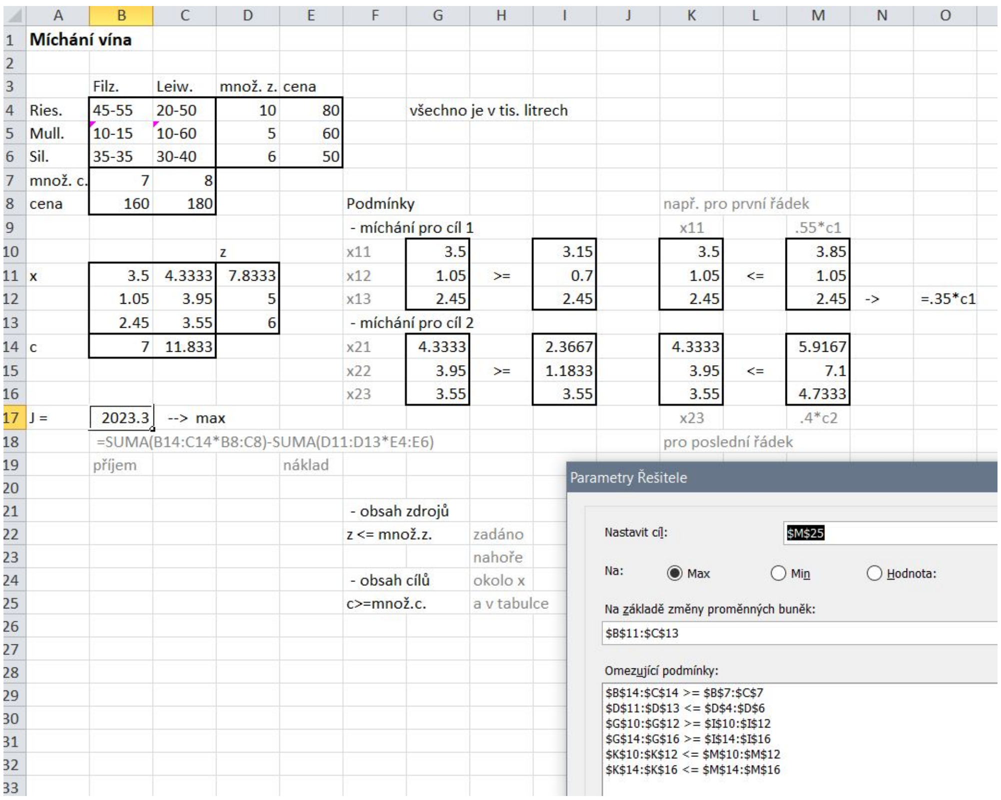
Podmínky na obsah zdrojových vín

$$z_1 \leq 10000,\ z_2 \leq 5000,\ z_3 \leq 6000$$

Podmínky na obsah cílových vín

$$c_1 \geq 7000,\ c_2 \geq 8000$$

Excel (U12_michaniVina.xlsx)



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	Míchání vína														
2															
3		Filz.	Leiw.	množ. z.	cena										
4	Ries.	45-55	20-50	10	80		všechno je v tis. litrech								
5	Mull.	10-15	10-60	5	60										
6	Sil.	35-35	30-40	6	50										
7	množ. c.	7	8												
8	cena	160	180			Podmínky					např. pro první řádek				
9						- míchání pro cíl 1					x11		.55*c1		
10				z		x11	3.5		3.15		3.5		3.85		
11	x	3.5	4.3333	7.8333		x12	1.05	>=	0.7		1.05	<=	1.05		
12		1.05	3.95	5		x13	2.45		2.45		2.45		2.45	->	=.35*c1
13		2.45	3.55	6		- míchání pro cíl 2									
14	c	7	11.833			x21	4.3333		2.3667		4.3333		5.9167		
15						x22	3.95	>=	1.1833		3.95	<=	7.1		
16						x23	3.55		3.55		3.55		4.7333		
17	J =	2023.3	--> max								x23		.4*c2		
18		=SUMA(B14:C14*B8:C8)-SUMA(D11:D13*E4:E6)									pro poslední řádek				
19		příjem			náklad										
20															
21						- obsah zdrojů									
22						z <= množ.z.		zadáno							
23								nahoře							
24						- obsah cílů		okolo x							
25						c>=množ.c.		a v tabulce							

2.4 Minimální tok nákladů v síti (MCNF)

Minimum Cost Network Flow Problem

Obecná formulace úlohy

Je dána síť (souvislý, orientovaný, acyklický graf s jedním vstupem a jedním výstupem) s m uzly a n hranami. Písmenem b_j označíme zásobu v uzlu j, tj. "výstupní tok" - "vstupní tok", tj . Jestliže je $b_j > 0$ jedná se o zásobovací uzel (zdroj), pro $b_j < 0$ jde o uzel s požadavkem (cíl) a pro $b_j = 0$ máme uzel průjezdní. Při tom pro uzel j je

$$\text{výstupní tok} = \textstyle\sum_k x_{jk} \text{ a vstupní tok } \sum_i x_{ij}.$$

S každou hranou je spojena dolní mez L_{ij} a horní mez U_{ij} toku touto hranou. Cílem je určit velikosti toků x_{ij} v jednotlivých hranách grafu tak, aby náklady na přepravu byly minimální, jestliže jednotková cena transportu po hraně ij je c_{ij}.

Předpokládáme, že platí $\sum_i b_i = 0$ (vyrovnané zdroje a požadavky).

Řešení

Pomocí LP

$$\sum_{ij} c_{ij} x_{ij} \to \min$$

$$\sum_k x_{jk} - \sum_i x_{ij} = b_j, \ \ \forall j$$

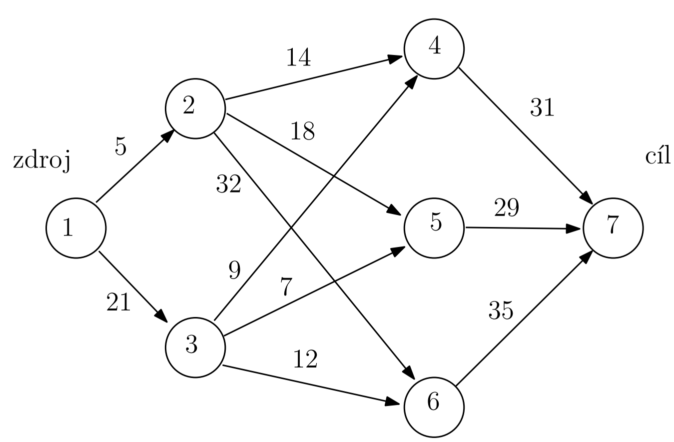
$$x_{ij} \leq U_{ij}, \ \ x_{ij} \geq L_{ij}, \ \ 0 \leq L_{ij} \leq U_{ij}, \ \ \forall i, j$$

Poznámka

c a x jsou čtvercové matice kde řádky i sloupce jsou indexovány uzly. Prvky těchto matic odpovídají hranám grafu. Většinou ne všechny hrany existují. Neexistující hrany vyplníme nulami. Jako měnící se buňky v Excelu zadáme jen existující hrany - nejlépe vytažené mimo do vektoru. Pro kritérium i podmínky průtoku můžeme použít celé matice (i prvky odpovídající neexistujícím hranám - jsou to nuly). Podmínky průtoku konstruujeme jakoby pro diagonálu matice x a děláme sum(řádek)-sum(sloupec), kde v řádcích fixujeme (F4) písmenka a ve sloupcích čísla adres. Pak lze kopírovat. Vše je vidět v Excelu.

2.4.1 Příklad

Přeprava zboží po síti drah.
Ze skladu (1) potřebujeme dovést zboží do výrobního závodu (7). K výrobě potřebujeme 150 výrobků a můžeme je poslat po drahách vyznačených v grafu pomocí šipek. U hran grafu jsou vyznačeny jejich vzdálenosti, které jsou úměrné ceně za přepravu



Každá dráha je kapacitně omezena na přepravu maximálně 80 ks zboží. Jak (po jakých drahách) je třeba zboží přepravit, aby náklady za přepravu byly minimální?

Řešení

Nejprve sestavíme **matici** c **délek hran** - $c_{i,j}$ jsou vzdálenosti z uzlu i do uzlu j podle zadaného grafu.

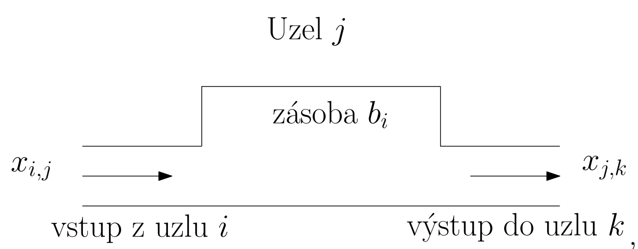
c j / i	1	2	3	4	5	6	7
1	1000	5	21	1000	1000	1000	1000
2	1000	1000	1000	14	18	32	1000
3	1000	1000	1000	9	7	12	1000
4	1000	1000	1000	1000	1000	1000	31
5	1000	1000	1000	1000	1000	1000	29
6	1000	1000	1000	1000	1000	1000	35
7	1000	1000	1000	1000	1000	1000	1000

Poznámka

Matici délek hran jsme zvolili v tzv penalizačním tvaru - tj. hranám, ke kterým neexistují dráhy, jsme přiřadili velké vzdálenosti tak, aby nemohly být v optimalizaci vybrány. Tento krok výrazně zjednodušuje zadání úlohy do Excelu, ale pracuje s mnohem větším počtem proměnných, než kdybychom jako stav zvolili jen platné hrany (se skutečnými vzdálenostmi). Protože nám jde především o zadávání příkladů, budeme tuto metodu penalizace využívat.

Stavové veličiny x zvolíme ve tvaru matice se stejnými rozměry jako má matice délek. $x_{i,j}$ bude počet jednotek zboží přepravených z uzlu i do uzlu j. Řádky i sloupce odpovídají uzlům 1,2,$\cdots$,7.

Řešení budeme zadávat pro toky jednotlivými uzly (ty si můžeme položit jakoby do diagonály stavové matice). Každý uzel si můžeme představit jako na obrázku (zde uvažujeme jen jeden vstup a jeden výstup)



tedy z předchozího uzlu i přichází tok $x_{i,j}$ a ven, do uzlu k, odchází tok $x_{j,k}$. Rozdíl „výstup" - „vstup" je zásoba $b_i = x_{j,k} - x_{i,j}$. Je-li vstup větší než výstup, bude $b_i > 0$ a jedná se o uzel zdrojový - dodává zboží ze své zásoby do sítě. Pro $b_i < 0$ jde o uzel spotřební - odebírá zboží ze sítě do své zásoby. Pro $b_i = 0$ se jedná o uzel tranzitní - zboží zde jen prochází.

Obecně ale do uzlu j může vstupovat a vystupovat více uzlu. Potom platí, že zásoba je to, co všechno z uzlu vyteče minus to, co všechno do uzlu vteče.

Z uzlu j vytéká $\sum_k x_{jk}$ a vtéká $\sum_i x_{ij}$. Tedy zásoba bude

$$\sum_k x_{jk} - \sum_i x_{ij} = b_j$$

Slovně: Zásoba v uzlu j je součet prvků x v j-tém řádku minus součet v j-tém sloupci.

V našem příkladě, kdy chceme převést 150 jednotek zboží z uzlu (1) do uzlu (7) bude pro vektor zásob b platit:

1. uzel (1) je zdrojový se zásobou 150, tedy $b_1 = 150$.
2. uzel (7) je spotřební s požadavkem 150, tedy $b_7 = -150$
3. ostatní uzly budou průchozí, $b_j = 0,\ j = 2, 3, \cdots, 6$

Podmínky na uzly budou

$$\sum_k x_{jk} - \sum_i x_{ij} = b_j,\ \ j = 1, 2, \cdots, 7$$

Poznámka *k realizaci v Excelu*

Jak jsme se již zmínili, polohu uzlů můžeme umístit na diagonálu matice x. Rozdíly součtů řádků a sloupců můžeme umístit do buněk, pod buňkami na diagonále x. Poloha buňky potom určuje řádek a sloupec, který do výpočtu vstupuje.

Z počítaných buněk stačí konstruovat jen první a zbytek kopírovat. Proto se ale buňky musí vhodně fixovat. Součet v řádku se nesmí posouvat v písmenech - tedy musí se fixovat písmena ($ před písmena v adrese). Tomu odpovídá 3x zmáčknout F4. Součet sloupce se nesmí posouvat v číslech - fixujeme čísla ($ před čísly v adrese) což dosáhneme, když 2x zmáčkneme F4 (ihned po zadání adresy).

Při větší úloze se pak zadávají zkonstruované buňky (v diagonální poloze) rovny odpovídajícím buňkám s b. Pro lepší zadání do řešitele je dobré zkonstruované buňky zkopírovat do sloupce podél sloupce b. !!! Pozor, nelze kopírovat tažením myší - některé směry jsou fixované !!! Je potřeba do každé buňky nového sloupce napsat „=odkaz na kopírovanou buňku“. Do podmínek řešitele se pak zadá: vlevo nový zkopírovaný sloupec, vpravo vektor b.

Další podmínky budou $x \leq 80$, kde 80 je maximální možný tok hranami.

Kriterium

$$J = \sum_{i,j} c_{i,j} x_{ij} \to \min,$$

tj. minimální vzdálenost (cena) při převozu.

Úloha LP

x je matice přenosů z uzlu i do uzlu j (matice7×7)

Kriterium

$$J = \sum_{i,j} c_{ij} x_{ij} \to \min$$

Podmínky

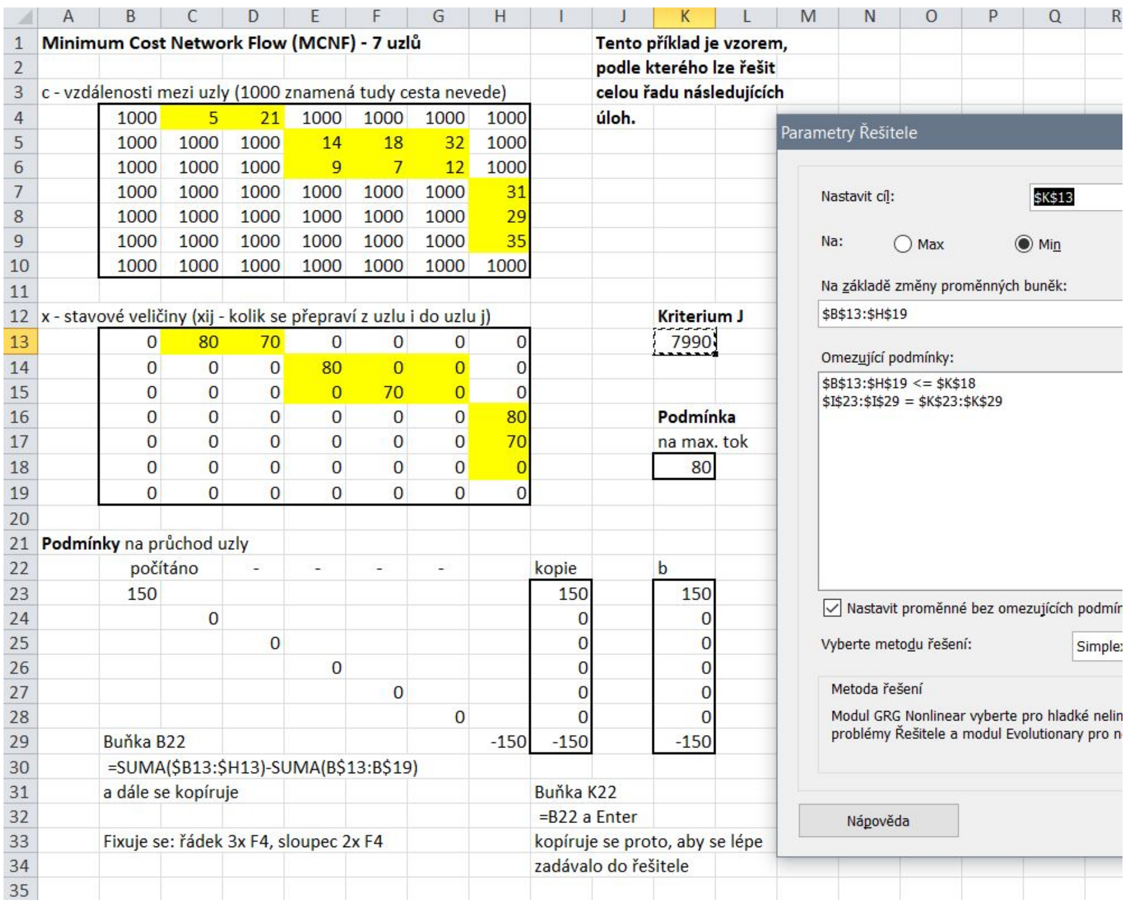
$$\sum_k x_{jk} - \sum_i x_{ij} = b_j,\ j = 1, 2, \cdots, 7$$

$$x_{ij} \leq 80,\ i, j = 1, 2, \cdots, 7$$

Poznámka

Protože jsme matici c (cen za převoz) zavedli v penalizačním tvaru, můžeme pracovat s celými maticemi c a x. Jinak bychom museli ceny i stavy vyzobávat z matic, což je při realizaci nepříjemné.

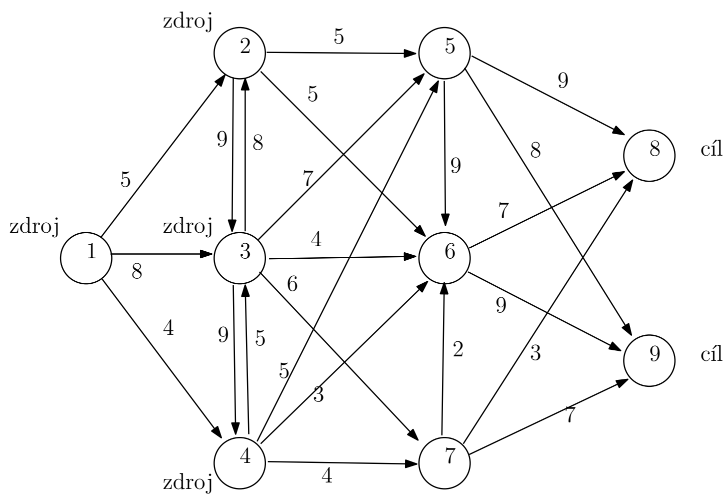
Excel (U15_MinimCesta_penal.xlsx)



Příklad 2.4.2 Přeprava zboží se čtyřmi zdroji a dvěma cíli.

Tento příklad přímo navazuje na předchozí. Tam, kde jsou zdroje, zadáme do vektoru b kladné hodnoty zásob, kde jsou cíle, zadáme záporné hodnoty požadavků.

Je dáno 9 uzlů podle obrázku



Ohodnocení (cena) hran grafu (drah mezi místy - uzly) je uvedeno v následující tabulce

	1	2	3	4	5	6	7	8	9
1	·	5	8	4	·	·	·	·	·
2	·	·	9	·	5	5	·	·	·
3	·	8	·	9	7	4	6	·	·
4	·	·	5	·	5	3	4	·	·
5	·	·	·	·	·	9	·	9	8
6	·	·	·	·	·	·	·	7	9
7	·	·	·	·	·	2	·	3	7

Jsou zde 4 zdroje, uzly 1 - 4. Požadavky jsou v uzlech 8 a 9. Zásoba ve zdrojových uzlech je

číslo uzlu	1	2	3	4
zásoba	50	30	20	50

a požadavky v cílových uzlech jsou

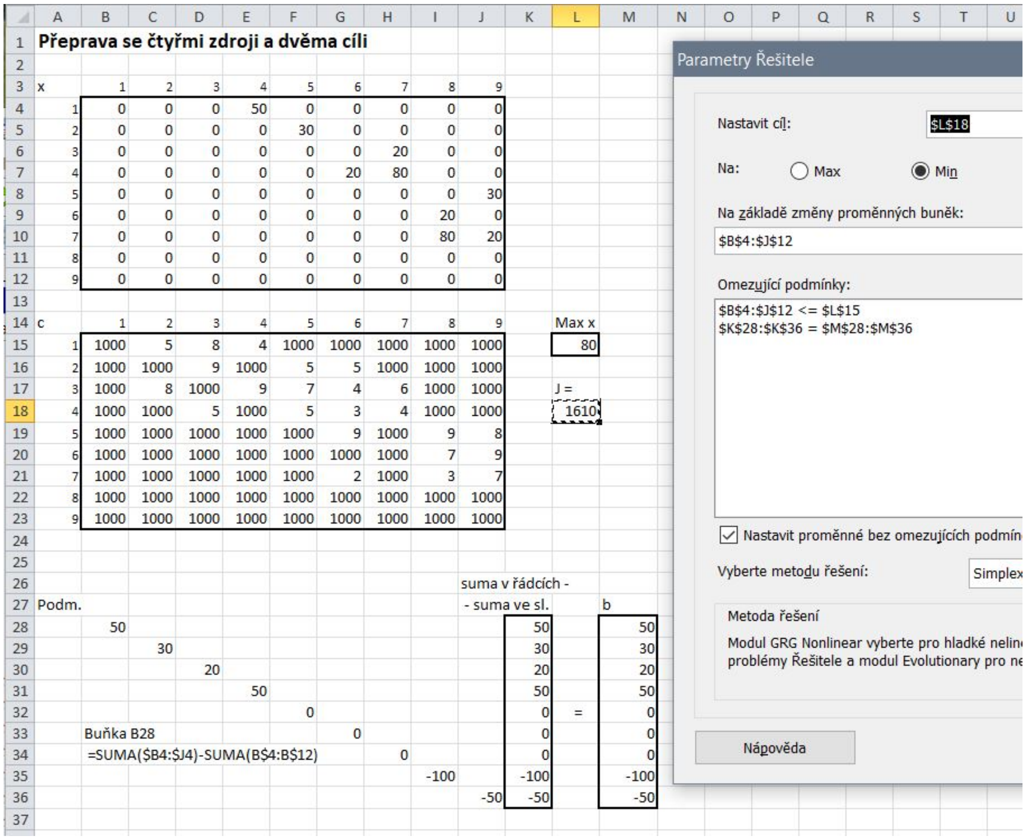
číslo uzlu	8	9
požadavek	100	50

.

Řešení

Řešení dostaneme přímo jako MCNF s maticí x stavových proměnných proměnných dimenze 9×9 a maticí cen c stejného rozměru s buňkami, ke kterým neexistují dráhy, doplněnými penalizací 1000. Řešení je patrné z Excelu, s příslušným řešitelem (viz předchozí vzorový příklad).

Excel (U16_MinimCesta2.xlsx)



2.5 Dopravní problém

Obecné zadání:

Je dáno m zdrojových uzlů, každý s z_i, kde $i \in \{1 \cdots m\} = M$ jednotkami zboží, a n cílových uzlů, každý s d_j, $j \in \{1 \cdots n\} = N$ jednotkami požadavků. c_{ij}, $i \in M$, $j \in N$ je jednotková cena za převoz zboží po hraně i, j a x_{ij} je množství zboží přepravené po této hraně. Chceme dosáhnout minima nákladů za převoz.

Řešíme pro pět zdrojů a šest cílů s maticí vzdáleností

$$c = \begin{bmatrix} 10 & 12 & 9 & 15 & 18 & 21 \\ 6 & 8 & 17 & 9 & 15 & 17 \\ 22 & 15 & 14 & 18 & 16 & 14 \\ 14 & 18 & 22 & 21 & 19 & 25 \\ 31 & 15 & 14 & 18 & 16 & 14 \end{bmatrix}$$

Zásoby dodavatelů jsou

$$z = [30, 20, 50, 10, 40]'$$

a požadavky odběratelů

$$d = [20, 30, 10, 10, 60, 20]'.$$

Řešení (klasicky)
Stavové veličiny zavedeme jako matici x stejných rozměrů, kde prvek $x_{i,j}$ znamená množství zboží, převezené z uzlu i do uzlu j.
Kriterium je

$$J = \sum_{i,j} c_{ij} x_{ij} \to \min$$

Podmínky vyjadřují splnění požadavků odběratelů a dodavatelů.

Součet prvků prvního sloupce matice x je $x_{1,1} + x_{2,1} + \cdots$, tedy říká, kolik dostal první odběratel dohromady od všech dodavatelů. Podobně součet druhého sloupce x je zboží, které dostal druhý odběratel. Stejně i pro ostatní sloupce. Tato množství porovnáme s požadavky odběratelů, tedy

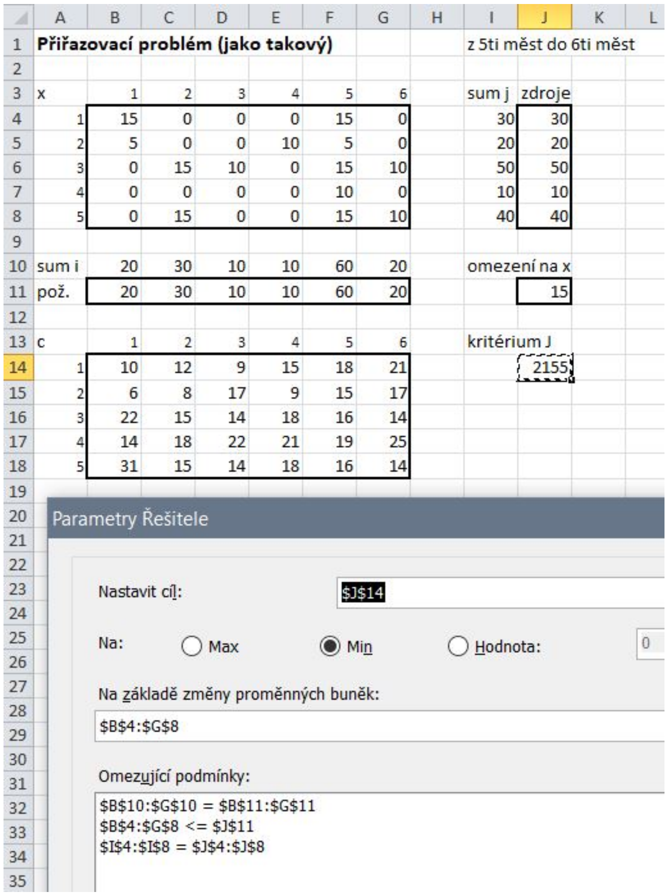
$$\sum_i x_{i,j} = d_i, \;\; j = 1, 2, \cdots, 6$$

Podobně, součty řádků matice x představují množství zboží, které dodal příslušný dodavatel celkem různým odběratelům. Ty opět porovnáme se zásobami dodavatelů, tedy

$$\sum_j x_{i,j} = z_i, \;\; i = 1, 2, \cdots, 5$$

Navíc můžeme stanovit maximální množství zboží, přenesené po hraně. Tady ji nastavíme na hodnotu 15.

Excel (U17_prirazProbl.xlsx)



Řešení (pomocí MCNF)

Tuto úlohu je možno řešit také jako Minimální tok nákladů v síti (viz Příklad 2.4.2).

Za tím účelem uvažujeme síť s 11 uzly - prvních 5 budou dodavatelé a zbylých 6 odběratelé. Podle předchozího zadání dostaneme:

– **matici vzdáleností** c

$$c = \begin{bmatrix} 1000 & 1000 & 1000 & 1000 & 1000 & 10 & 12 & 9 & 15 & 18 & 21 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 6 & 8 & 17 & 9 & 15 & 17 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 22 & 15 & 14 & 18 & 16 & 14 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 14 & 18 & 22 & 21 & 19 & 25 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 31 & 15 & 14 & 18 & 16 & 14 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \end{bmatrix}$$

Matice vzdáleností je v penalizační formě.

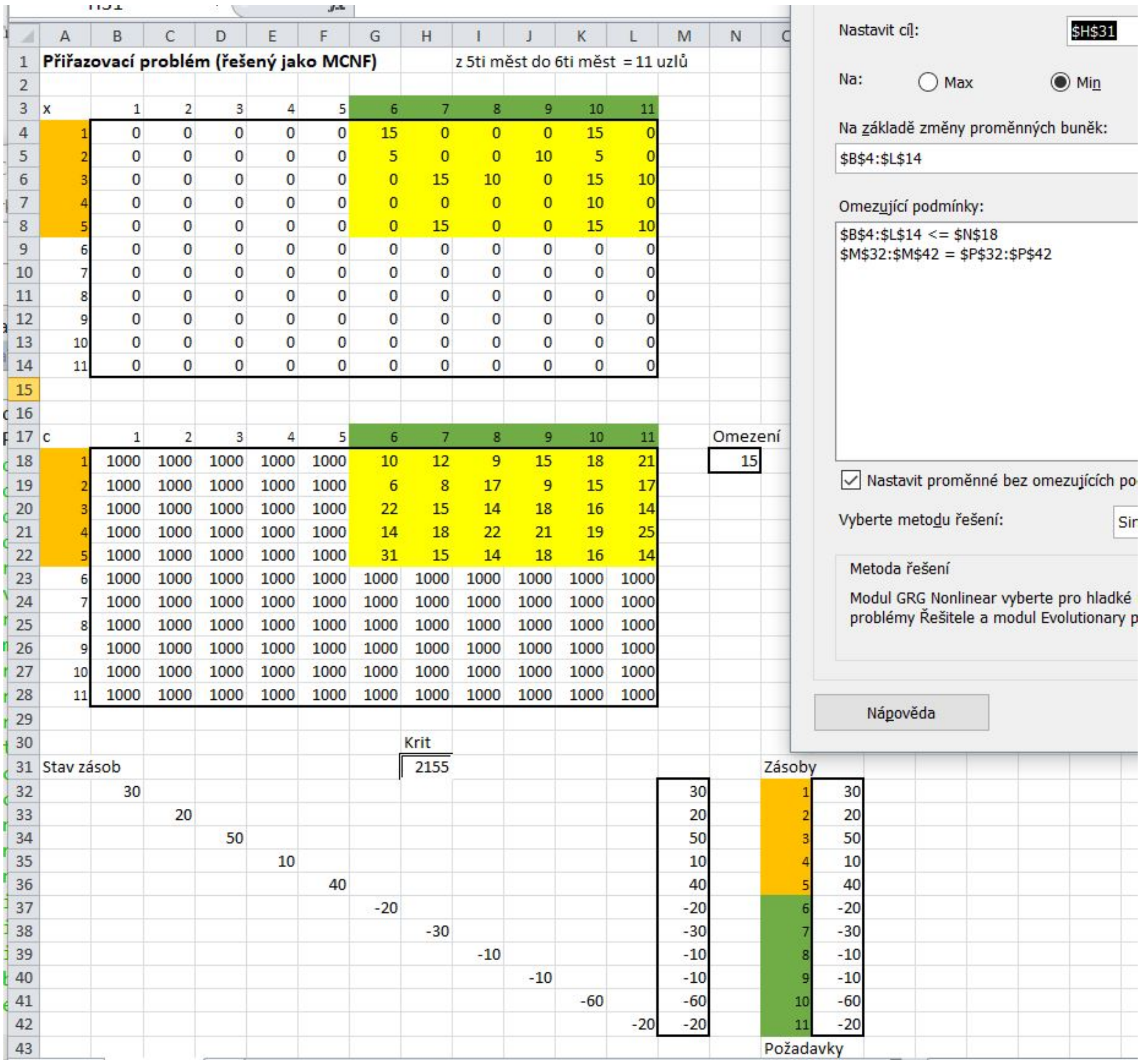
– **vektor** b

$$b = [30, 20, 50, 10, 40, -20, -30, -10, -10, -60, -20]'$$

kde kladné hodnoty jsou zásoby dodavatelů a záporné požadavky odběratelů.

– **maximum přenosu** po hranách bude opět 15.

Excel (U19_prirazProblMCNF_penal.xlsx)
vychází z MCNF - U15_MinimCesta_penal.xlsx



x	1	2	3	4	5	6	7	8	9	10	11
1	0	0	0	0	0	15	0	0	0	15	0
2	0	0	0	0	0	5	0	0	10	5	0
3	0	0	0	0	0	0	15	10	0	15	10
4	0	0	0	0	0	0	0	0	0	10	0
5	0	0	0	0	0	0	15	0	0	15	10
6	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0

c	1	2	3	4	5	6	7	8	9	10	11
1	1000	1000	1000	1000	1000	10	12	9	15	18	21
2	1000	1000	1000	1000	1000	6	8	17	9	15	17
3	1000	1000	1000	1000	1000	22	15	14	18	16	14
4	1000	1000	1000	1000	1000	14	18	22	21	19	25
5	1000	1000	1000	1000	1000	31	15	14	18	16	14
6	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
7	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
8	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
9	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
10	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000
11	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000	1000

Stav zásob											M		Zásoby
30											30	1	30
	20										20	2	20
		50									50	3	50
			10								10	4	10
				40							40	5	40
					-20						-20	6	-20
						-30					-30	7	-30
							-10				-10	8	-10
								-10			-10	9	-10
									-60		-60	10	-60
										-20	-20	11	-20

2.6 Problém přiřazení

Máme m pracovníků a n úkolů. Dále je stanovena výkonnost v_{ij} i-tého pracovníka vzhledem k úkolu j (úkoly jsou různého druhu a pracovníci mají různé specializace). Požadujeme splnění všech úkolů s minimálním výkonem (cenou) pracovníků při dalších vedlejších podmínkách.

Poznámka

U řešení bychom přirozeně požadovali celočíselnost. To ale vede na tzv. celočíselné programování, kterým se budeme zabývat až v následujícím kurzu LP2. Zatím se spokojíme s řešením reálným, které vhodně zaokrouhlíme.

2.6.1 Příklad

Výběr z investičních příležitostí.

Firma má k dispozici 3 mil. korun na investice a dále má možnost si půjčit 1 mil. korun od banky na 12% roční úrok. K dispozici jsou následující investiční příležitosti

Typ investice	Očekávané procento ročního zisku	Očekávané procento zhodnocení majetku	Procento risku/Kč
1. Nemovitosti	0	18	20
2. Stříbro	0	10	12
3. Spořící účet	2	0	1
4. Blue chip zásoby	3	6	7
5. Dluhopisy	4	0	3
6. Hi-tech zásoby	0	20	30

Cílem návrhu je maximum majetku po jednom roce při následujících podmínkách:

1. Očekávané zhodnocení majetku musí být větší než 7%.
2. Nejméně 50% z investovaných financí musí jít do zásob a dluhopisů.
3. Maximálně 20% z vlastních financí (ne půjčených) může jít do nemovitostí a stříbra.
4. Průměrný risk celé akce nesmí překročit 10.

Řešení

Jako optimalizovaný stav zvolíme $x_1 \cdots x_6$ - množství peněz, investovaných do jednotlivých akcí $1, \cdots, 6$ a jako poslední x_7 bude množství peněz, které si půjčíme u banky.

Celkový majetek po jednom roce bude dán tím, co jsme měli na začátku (1 mil.) + to, co jsme vydělali (tabulka) - úroky z půjčky. Přitom to , co jsme vydělali určíme jako procento z investice podle tabulky - buď jako roční zisk, nebo zhodnocení majetku, tedy v procentech investice

akce	1	2	3	4	5	6
procenta	18	10	2	3+6=9	4	20

přičemž investujeme vlastní i půjčené. Půjčené + úrok musím vrátit. kritérium tedy bude

$$1.18x_1 + 1.10x_2 + 1.02x_3 + 1.09x_4 + 1.04x_5 + 1.20x_6 - 1.12x_7 \rightarrow \max$$

Podmínky

1. Můžeme investovat jen to, co máme

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 \leq 3\,\text{mil.}$$

2. Půjčit si můžeme maximálně 1 mil.

$$x_7 \leq 1\,\text{mil.}$$

3. Přinejmenším 7% zhodnocení investovaného vlastního vkladu

$$1.18x_1 + 1.10x_2 + x_3 + 1.06x_4 + x_5 + 1.20x_6 \geq 1.07\,(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$$

4. Alespoň polovina peněz na zásoby a dluhopisy

$$x_4 + x_5 + x_6 \geq 0.5\,(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$$

5. Maximálně 20% z vlastních investic do nemovitostí a stříbra

$$x_1 + x_2 \leq 0.2\,(3\,\text{mil.})$$

6. Průměrný risk investice

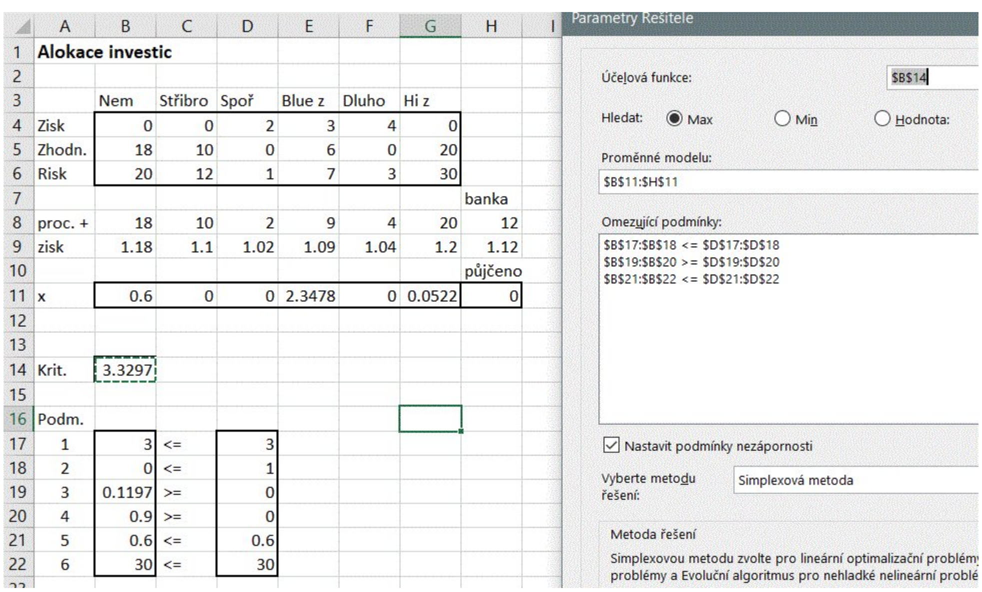
$$20x_1 + 12x_2 + x_3 + 7x_4 + 3x_5 + 30x_6 \leq 10\,(x_1 + x_2 + x_3 + x_4 + x_5 + x_6)$$

7. Nezápornost

$$x_1, \cdots, x_7 \geq 0$$

Tento příklad ještě porovnáme se situací, kdy úrok z půjčky bude 10%.

Excel: (U20_vyberInvestic.xlsx)



Pokračování příkladu

Vycházíme z předchozího příkladu, ale jako kritérium máme minimalizaci rizika.

Řešení

kritérium

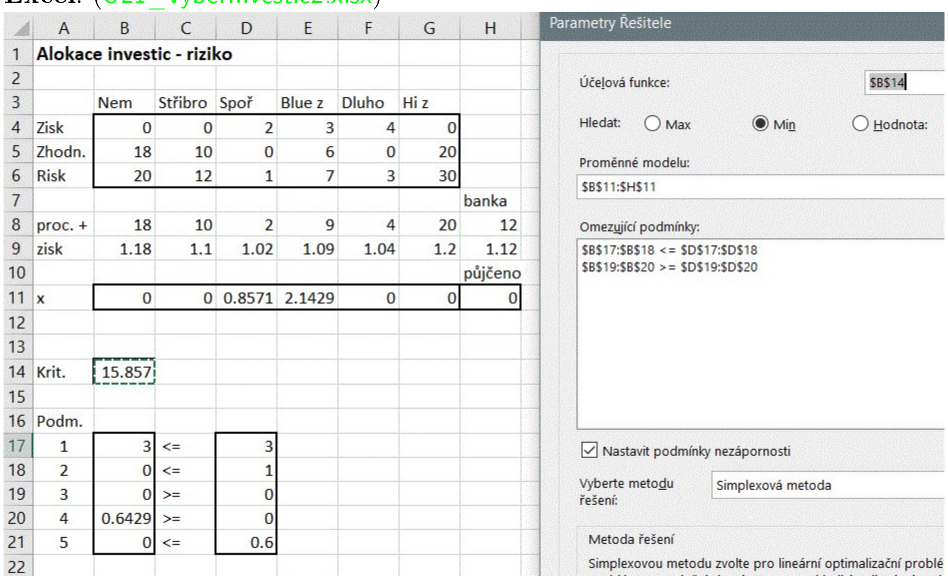
$$20x_1 + 12x_2 + x_3 + 7x_4 + 3x_5 + 30x_6 \to \min$$

Dále upravíme podmínky:

- podmínku na riziko vynecháme,

- v podmínce na zhodnocení majetku místo investované částky (suma x_i) dosadíme celou částku (3 mil.) (Víme proč?)

Excel: (U21_vyberInvestic2.xlsx)



	A	B	C	D	E	F	G	H
1	Alokace investic - riziko							
2								
3		Nem	Stříbro	Spoř	Blue z	Dluho	Hi z	
4	Zisk	0	0	2	3	4	0	
5	Zhodn.	18	10	0	6	0	20	
6	Risk	20	12	1	7	3	30	
7								banka
8	proc. +	18	10	2	9	4	20	12
9	zisk	1.18	1.1	1.02	1.09	1.04	1.2	1.12
10								půjčeno
11	x	0	0	0.8571	2.1429	0	0	0
12								
13								
14	Krit.	15.857						
15								
16	Podm.							
17	1	3	<=	3				
18	2	0	<=	1				
19	3	0	>=	0				
20	4	0.6429	>=	0				
21	5	0	<=	0.6				
22								

2.6.2 Příklad

Přiřazení policejních hlídek do oblastí.

V určitém městě s pěti oblastmi máme rozmístit policejní hlídky. Oblasti jsou různě nebezpečné, proto každá z nich vyžaduje jiný minimální počet členů hlídky. Dále je v oblastech definován tzv. koeficient stupně ochrany. Účinnost policie dostaneme, když počet členů hlídky násobíme tímto koeficientem. Např. má-li hlídka šest členů a koeficient je 3, pak účinnost je 6·3 = 18. Koeficienty minimální účinnosti v jednotlivých oblastech jsou v tabulce

oblast	1	2	3	4	5
koeficient	3	7	10	5	4
účinnost	40	50	70	60	60

Chceme optimalizovat (minimalizovat náklady) počet členů hlídek za následujících podmínek

1. Musí být zachována minimální účinnost.

2. Průměrná účinnost musí být alespoň 50.

3. Pro oblasti 1, 2 a 3 je třeba alespoň o 50% více policistů než pro oblasti 4 a 5.

K dispozici je 67 domácích policistů, další je třeba najmout s platem 1000 Kč/den.

Řešení
Optimalizovaný stav budou tvořit počty členů hlídek pro jednotlivé oblasti

$$x = [x_1, x_2, x_3, x_4, x_5] .$$

kritérium bude vyjadřovat náklady na najaté policisty

$$1000\,(x_1 + x_2 + x_3 + x_4 + x_5 - 67) \rightarrow \min$$

Podmínky jsou

1. Budou využiti všichni domácí policisté

$$x_1 + x_2 + x_3 + x_4 + x_5 \geq 67$$

2. Bude dodržena minimální účinnost

$$\begin{aligned} 3x_1 &\geq 40 \\ 7x_2 &\geq 50 \\ 10x_3 &\geq 70 \\ 5x_4 &\geq 60 \\ 4x_5 &\geq 40 \end{aligned}$$

3. Průměrná účinnost

$$(3x_1 + 7x_2 + 10x_3 + 5x_4 + 4x_5)\,/5 \geq 50$$

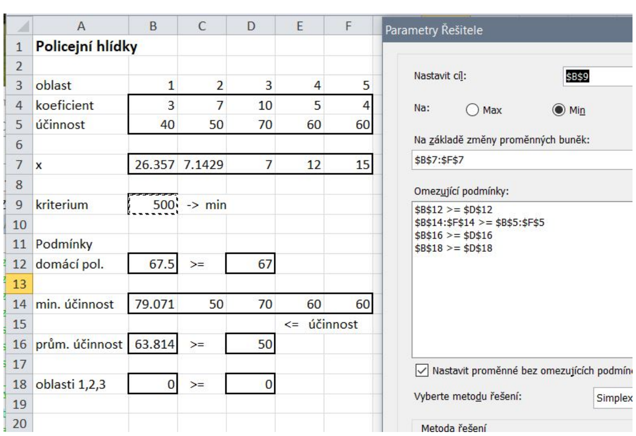
4. Více policistů pro oblasti 1, 2 a 3

$$x_1 + x_2 + x_3 \geq 1.5\,(x_4 + x_5)$$

5. Nezápornost

$$x \geq 0$$

Excel: (U22_policejniHlidky.xlsx)



2.6.3 Příklad

Tvorba rozvrhu.

Student plánuje rozvrh na příští semestr. Předměty a počty bodů k úspěšnému absolvování jsou v tabulce

	Marketing	Podnikání	Účetnictví	Matematika	Finance
Body/hodinu	5	4.5	5.5	3.5	5.5
Požadavek	50	55	60	50	50

Další podmínky

1. Je třeba, aby úspěšně absolvoval všechny kurzy.
2. Semestr nabízí maximálně 15 hodin z každého předmětu.
3. Průměrné ohodnocení musí být alespoň 64 bodů (tj. celkem 320 bodů).
4. Počet hodin z Matematiky musí být alespoň 20% z hodin ostatních předmětů.
5. Student si vydělává u McDonalda. Dostává 300 Kč na hodinu. Pro návštěvu kurzů a výdělek má za semestr k dispozici 800 hodin (zbytek si vyhradil na studium). Pro obživu potřebuje alespoň 30000 Kč na celý semestr.

Kolik hodin si student má alokovat na jednotlivé kurzy, aby splnil všechny podmínky a zbylo mu co nejvíce času na práci pro obživu?

Řešení

Stav x bude představovat počet alokovaných hodin pro jednotlivé předměty.

Kritérium

$$x_1 + x_2 + x_3 + x_4 + x_5 \to \min$$

$\cdots$ čím méně stráví času ve skole, tím více mu zbude na práci.

Podmínky

podm. 1: splnění bodů za kurzy

$$5x_1 \geq 50$$

$$4.5x_2 \geq 55$$

$$5.5x_3 \geq 60$$

$$3.5x_4 \geq 50$$

$$5.5x_5 \geq 50$$

podm. 2: nabídka 15 kurzů

$$x_i \leq 15,\ i = 1, 2, \cdots, 5$$

podm. 3: průměrné hodnocení

$$5x_1 + 4.5x_2 + 5.5x_3 + 3.5x_4 + 5.5x_5 \geq 320$$

podm. 4: hodiny z matematiky

$$x_4 \geq 0.2\,(x_1 + x_2 + x_3 + x_5)$$

podm. 5: McDonald

$$300\,(800 - x_1 - x_2 - x_3 - x_4 - x_5) \geq 30000$$

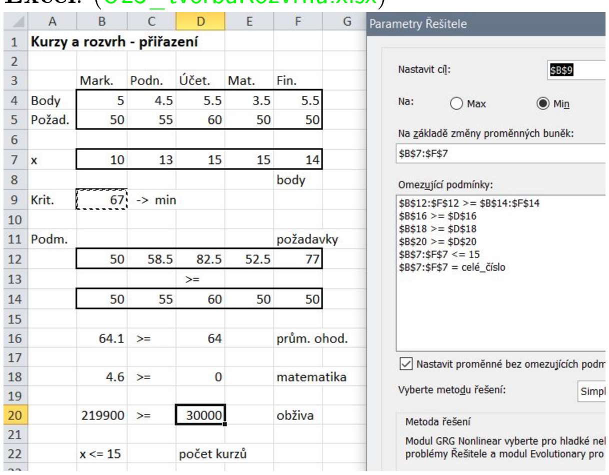
$\cdots$ vnitřek závorky představuje hodiny, které zbudou na práci.

Nezápornost

$$x \geq 0$$

Excel

Excel: (U23_tvorbaRozvrhu.xlsx)



2.7 Nejkratší dráha grafem (pomocí MCNF)

Jedná se o typickou úlohu řešenou v teorii grafů.

Je dána ohodnocená síť, kde ohodnocení c_{ij} interpretujeme jako délky jednotlivých hran (i, j). Úkol je najít nejkratší dráhu ze zdrojového uzlu 1 do cílového uzlu m.

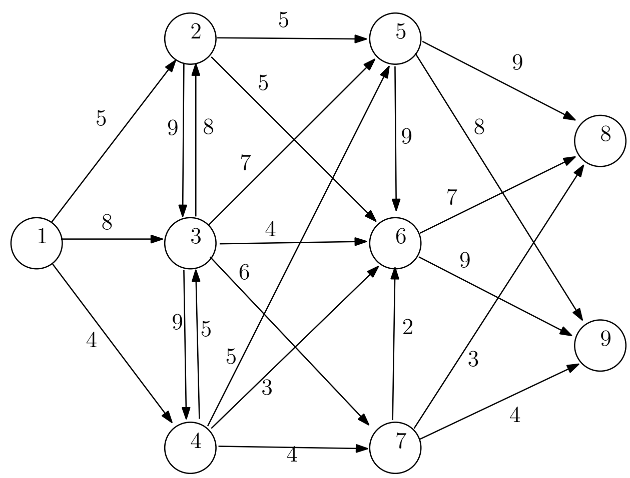
Řešíme pomocí MCNF, kde uvažujeme o přepravě jednotky toku z uzlu 1 do uzlu m. Horní mez pro jednotlivé toky položíme $U_{ij} = 1,\ \forall ij$ a všechny uzly zavedeme jako průtokové $b_i = 0,\ \forall i$.

2.7.1 Příklad

Nejkratší dráha grafem pomocí MCNF.

Je dána ohodnocená síť, kde ohodnocení c_{ij} interpretujeme jako délky jednotlivých hran (i, j). Úkol je najít nejkratší dráhu ze zdrojového uzlu 1 do cílového uzlu m.

V našem příkladě uvažujeme graf z obrázku



Hledáme nejkratší dráhu z uzlu 1 do uzlu 7.

Řešení (MCNF)

Matice c přímých vzdáleností mezi uzly bude

$$c = \begin{bmatrix} 1000 & 5 & 8 & 4 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 9 & 1000 & 5 & 5 & 1000 & 1000 & 1000 \\ 1000 & 8 & 1000 & 9 & 7 & 4 & 6 & 1000 & 1000 \\ 1000 & 1000 & 5 & 1000 & 5 & 3 & 4 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 9 & 1000 & 9 & 8 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 7 & 9 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 2 & 1000 & 3 & 4 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 \end{bmatrix}$$

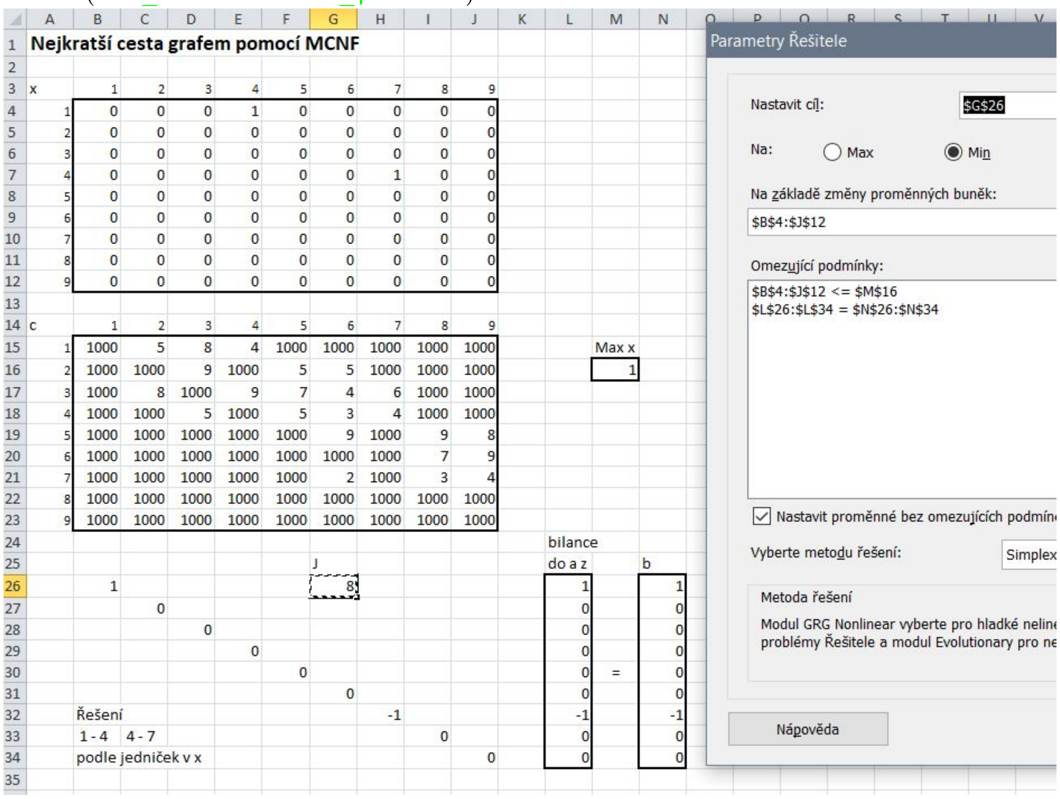
opět v penalizačním tvaru.

Stavová proměnná x bude matice stejných rozměrů. Její prvky $x_{i,j}$ představují dráhu z uzlu i do uzlu j.

Vektor b (zásoby v uzlech) zadáme tak, že do uzlu 1 (ze kterého hledáme dráhu) vložíme jednotu zboží, $b_1 = 1$ a do uzlu 7 kde má dráha končit zadáme požadavek na jednotku zboží $b_7 = -1$. Ostatní uzly ponecháme jako průchozí, $b_i = 0$.

Maximální kapacitu hran nastavíme na 1.

Excel (U25_minCestaMNCF_penal.xlsx)



x	1	2	3	4	5	6	7	8	9
1	0	0	0	1	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	1	0	0
5	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0

c	1	2	3	4	5	6	7	8	9
1	1000	5	8	4	1000	1000	1000	1000	1000
2	1000	1000	9	1000	5	5	1000	1000	1000
3	1000	8	1000	9	7	4	6	1000	1000
4	1000	1000	5	1000	5	3	4	1000	1000
5	1000	1000	1000	1000	1000	9	1000	9	8
6	1000	1000	1000	1000	1000	1000	1000	7	9
7	1000	1000	1000	1000	1000	2	1000	3	4
8	1000	1000	1000	1000	1000	1000	1000	1000	1000
9	1000	1000	1000	1000	1000	1000	1000	1000	1000

B	C	D	E	F	G	H	I	J	bilance do a z		b
1					8				1		1
	0								0		0
		0							0		0
			0						0		0
				0					0	=	0
					0				0		0
Řešení						-1			-1		-1
1 - 4	4 - 7						0		0		0
podle jedniček v x								0	0		0

2.8 Bankovní úvěr

Banka obchoduje s penězi. Půjčuje a za půjčení dostává úvěr. Podobně jako podniká s kapitálem je možno investovat do různých akcí ve formě půjček s různými úrokovými sazbami a dalšími podmínkami pro půjčování. Jak má banka s penězi zacházet, aby dosáhla maxima zisku z úroků?

2.8.1 Příklad

Bankovní půjčky s různými úroky.

Banka disponuje částkou 53 milionů US$, z nichž poskytuje pět druhů půjček s různými úrokovými sazbami:

a. obchodní úvěr	12.5%
b. hypoteční úvěr (prvotní)	13.7%
c. úvěr na údržbu domu	13.6%
d. hypoteční úvěr (druhotný)	14.3%
e. krátkodobý překlenovací úvěr	18.1%

S výjimkou krátkodobého překlenovacího úvěru, který je omezen na 3 miliony US$ nejsou půjčky ve své výši omezeny, musí však respektovat určitou regulaci úvěrových postupů:

1. úvěr na údržbu nesmí přesáhnout 20 % prvotních hypoték,

2. obchodní úvěr nesmí být větší než druhotné hypotéky,

3. banka musí investovat alespoň 60 % úvěrů do hypoték,

4. kvůli zabezpečení proti rizikům je nutno na každý dolar druhotných hypoték investovat alespoň dva dolary do prvotních hypoték.

Účelem je maximalizovat investiční výnos (výnos z úroků za úvěr).

Řešení

Stav $x = [x_1, x_2, x_3, x_4, x_4]$ - kolik se z čeho půjčí (a. b. c. d. e.).

Kriterium - maximální výnos z úroků

$$J = 12.5x_1 + 13.7x_2 + 13.6x_3 + 14.3x_4 + 18.1x_5 \to \max$$

Podmínky

1. kolik banka disponuje

$$\sum_i x_i \leq 53000$$

2. omezení na krátkodobý úvěr

$$x_5 \leq 3$$

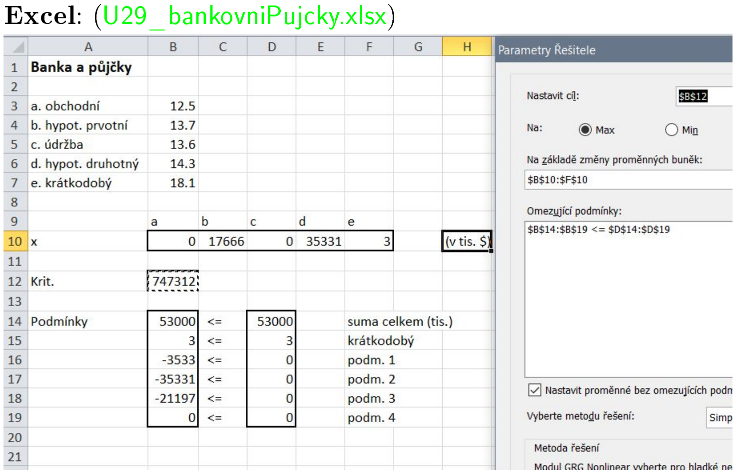
3. - 6 podle uvedených 1. - 4.

$$x_3 \leq 0.2x_2$$
$$x_1 \leq x_4$$
$$0.6\sum_{i=1}^{5} x_i \leq x_2 + x_4$$
$$x_4 \leq 2x_2$$

Poznámka

Do podmínek řešitele lze na pravé straně zadat pouze číslo neb odkaz na buňky. Tedy podmínky, kde je na pravé straně výpočet by se museli nejprve spočítat v listu a na výsledek zadat odkaz. Jednodušší řešení ale je převést pravou stranu podmínka nalevo a vpravo pak dosadit nulu. Např. místo $x_3 \leq 0.2x_2$ *zadat* $x_3 - 0.2x_2 \leq 0$. *Tak je to také zde realizováno.*

Excel: (U29_bankovniPujcky.xlsx)



2.9 Obsazení směn

V případě obchodu s rychlým občerstvením (ale i v řadě dalších případů, jako u zdravotních sester, pilotů, pokladních v samoobsluhách atd.) je zapotřebí různý počet zaměstnanců v různých denních dobách. Úlohu řešíme tak, že naplánujeme stejně dlouhé a překrývající se směny. Navrhujeme počet zaměstnanců pro jednotlivé směny a jejich překryv nám umožní plánovat různý počet zaměstnanců v různou denní dobu.

2.9.1 Příklad

Obsazení směn se 4-hodinovými intervaly.

V daném obchodě se pracuje s osmihodinovou směnou, která může začínat vždy po čtyřech hodinách: 6.00, 10.00, 14.00, 18.00, 22.00 a 02.00 hodin. Nároky na požadovaný počet zaměstnanců pro jednotlivé 4hodinové úseky jsou následující

i	1	2	3	4	5	6
úsek	06-10	10-14	14-18	18-22	22-02	02-06
počet	17	9	19	12	5	8

Vysvětlení: Ti, co nastoupí v 06h budou pracovat 8 hodin, tedy až do 14h, tj. oba dva po sobě následující 4hodinové úseky. Další směna nastoupí v 10h a bude pracovat do 18h. Tedy v úseku 10-14h budou pracovat ti, co nastoupili v 6h plus ti, co nastoupili v 10h. A tak je to s každým 4hodinovým úsekem.

Jak obsadit směny s minimálním počtem lidí?

Řešení

Zavedeme stav x se složkami x_i, $i = 1, 2, \cdots, 6$ - počet lidí, nastupujících na úsek i.

Model je

$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 \to \min$$

$$\begin{aligned} x_1 + x_6 &\geq 17 \\ x_1 + x_2 &\geq 9 \\ x_2 + x_3 &\geq 19 \\ x_3 + x_4 &\geq 12 \\ x_4 + x_5 &\geq 5 \\ x_5 + x_6 &\geq 8 \end{aligned}$$

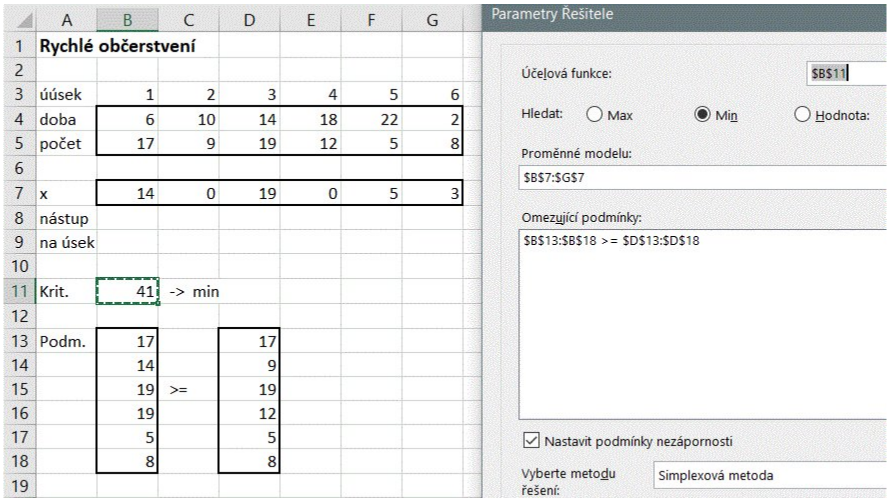
$$x_1, x_2, x_3, x_4, x_5, x_6 \geq 0 \ \ \text{(int)}$$

a řešíme.

Poznámka

x_i musí být celočíselné (nejde obsadit půl člověka). Jak jsme již řekli, zatím celočíselnost neumíme. Proto buď zaokrouhlíme nebo v řešiteli v okénku pro podmínky zadáme vlevo x a uprostřed vybereme „int” nebo „celé”. Co to znamená řekneme příští semestr.

Excel: (U26_smeny4hod.xlsx)



Modifikace příkladu

Totéž, ale úseky vezmeme dvouhodinové. Požadavky na úseky budou:

i	1	2	3	4	5	6	7	8	9	10	11	12
úsek	6-8	8-10	10-12	12-14	14-16	16-18	18-20	20-22	22-24	0-2	2-4	4-6
počet	17	11	9	7	1	19	12	8	5	3	3	8

Excel: (U26_smeny2hod.xlsx)

2.10 Dynamická produkce

V této třídě úloh se jedná o výrobu na odbyt, jehož požadavky přichází postupně v čase. Rychlost výroby je omezena a požadavky přicházejí v různých velikostech. Proto je výhodné tvořit zásoby. Za jejich uskladnění je ale potřeba platit.

2.10.1 Příklad

Plánování výroby na čtyři měsíce dopředu.

Plánujeme výrobu od začátku ledna do konce dubna - tedy na 4 měsíce. Údaje jsou v tabulce

	leden	únor	březen	duben
Požadavky (ks)	80	70	130	150
Kapacita výroby (ks)	120	140	150	140
Jednotkové náklady ($)	1	1.1	1.2	1.25

Stav zásob evidujeme vždy na začátku měsíce a označíme je $z_1, z_2, \cdots, z_5$. Hodnota z_1 je dána naší zásobou na začátku, tady $z_1 = 20$, na konci požadujeme hodnotu $z_5 = 0$. Jednotková cena za uskladnění je \$0.05 splatná na konci ledna, \$0.15 na konci února a března.

Jak vyrábět, abychom splnili požadavky a měli minimální náklady na výrobu a skladování?

Řešení

Pro zásoby bude platit: „zásoba teď” bude „zásoba minule” plus to, „co se vyrobilo” mínus to, „co se odevzdalo”. Tedy např. na začátku bude

$$z_2 = z_1 + v_1 - 80,$$

kde v_1, v_2, v_3, v_4 je výroba v jednotlivých měsících.

Zavedeme stav x ve tvaru

$$x = [v_1, v_2, v_3, v_4, z_2, z_3, z_4]$$

Kriterium

$$v_1 + 1.1v_2 + 1.2v_3 + 1.25v_4 + 0.05z_2 + 0.15z_3 + 0.15z_4 \to \min$$

$\cdots$ platíme náklady na výrobu a poplatky za skladování.

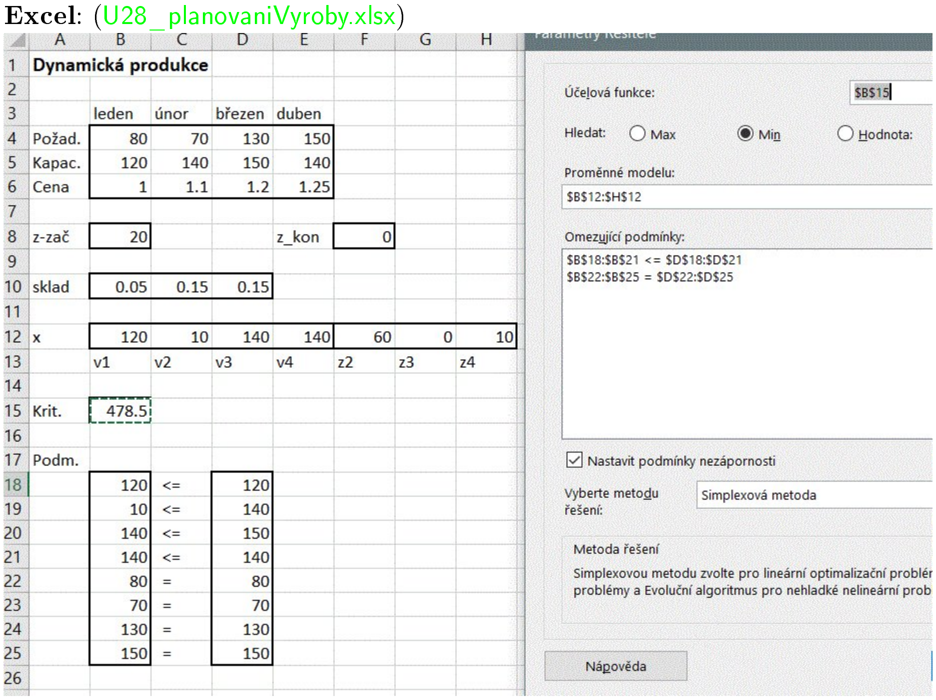
Podmínky

$$\begin{aligned} v_1 &\le 120 \\ v_2 &\le 140 \\ v_3 &\le 150 \\ v_4 &\le 140 \\ v_1 + z_1 - z_2 &= 80 \\ v_2 + z_2 - z_3 &= 70 \\ v_3 + z_3 - z_4 &= 130 \\ v_4 + z_4 - z_5 &= 150 \\ v_1, \cdots, v_4, z_2, \cdots, z4 &\ge 0 \end{aligned}$$

$\cdots$ na kapacitu a splnění požadavků.

Excel

Excel: (U28_planovaniVyroby.xlsx)



	A	B	C	D	E	F	G	H
1	**Dynamická produkce**							
2								
3		leden	únor	březen	duben			
4	Požad.	80	70	130	150			
5	Kapac.	120	140	150	140			
6	Cena	1	1.1	1.2	1.25			
7								
8	z-zač	20			z_kon	0		
9								
10	sklad	0.05	0.15	0.15				
11								
12	x	120	10	140	140	60	0	10
13		v1	v2	v3	v4	z2	z3	z4
14								
15	Krit.	478.5						
16								
17	Podm.							
18		120	<=	120				
19		10	<=	140				
20		140	<=	150				
21		140	<=	140				
22		80	=	80				
23		70	=	70				
24		130	=	130				
25		150	=	150				
26								

Dodatky k příkladu

Dále je možno uvažovat omezené kapacity skladu

$$z_2 \leq 40$$
$$z_3 \leq 50$$
$$z_4 \leq 50$$

S těmito omezeními je hodnota kriteria 479.50.

2.11 Řezání materiálu

Jedná se o typickou úlohu pro metody lineárního programování (s podmínkou celočíselnosti). Máme k dispozici určitý materiál (tyče o stejné základní délce) a z nich máme nařezat předepsaný počet kusů o daných délkách. Přitom většinou požadujeme minimální odpad. Variantou úlohy je případ, kdy můžeme řezat do zásoby.

Poznámka: O celočíselnosti jsme se již zmiňovali.

2.11.1 Příklad

Řezání nosníků.

K výrobě ocelových konstrukcí se používají nosníky o délce 1.7 m, 1.2 m a 0.5 m. Materiál potřebné kvality je dodáván pouze v délce 3.3 m.

a) Navrhněte takový způsob řezání materiálu, aby bylo nařezáno alespoň 600 kusů nosníků délky 1.7 m, alespoň 800 kusů délky 1.2 m a alespoň 800 půlmetrových kusů a celkový odpad (v m) byl minimální. Zásoba materiálu je dostatečná.

Řešení

Nejprve zvolíme tzv. řezné plány. Vybrané plány jsou v tabulce

plán	1	2	3	4	5	6
1.7	1	1	1	0	0	0
1.2	1	0	0	2	1	0
0.5	0	3	2	1	4	6
zbytek	0.4	0.1	0.6	0.4	0.1	0.3

Poznámky k řezným plánům

- *Čísla uvedená v tabulce jsou počty kusů určité délky, které při řezání podle daného plánu získáme.*
- *Zbytek je délka kusu nosníku, který po řezání zbude a nelze ho dále využít.*
- *Uvedené plány nejsou všechny možné, jen ty, které jsme z nějakého důvodu vybrali.*

Stav $x = [x_1, x_2, x_3, x_4, x_5, x_6]$ kde x_i je počet řezání podle plánu i.

Kritérium - minimální odpad

$$J = 0.4x_1 + 0.1x_2 + 0.6x_3 + 0.4x_4 + 0.1x_5 + 0.3x_6 \rightarrow \min$$

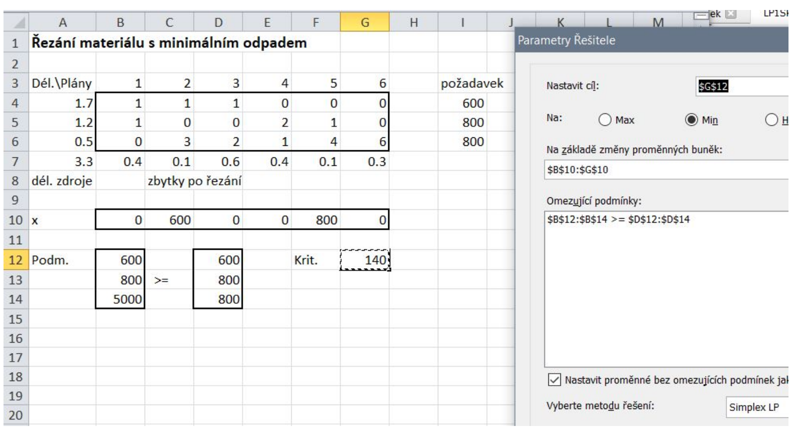
Podmínky - nařezané počty $\geq$ požadované (je-li tabulka $T_{i,j}$ a požadované počty jednotlivých délek d_i) pak

$$\sum_j T_{i,j} x_j \geq d_i, \quad i = 1, 2, 3$$

Poznámka

Všimněme si, že plán 3 je nesmyslný. Ze zbytku by se ještě mohla uříznout délka 0.5 - tím bychom získali plán 2 s lepším výsledkem. Je zřejmé, že ve výsledném řešení tento plán nesmí být vybrán. Z Excelu vidíme, že vybrán není.

Excel: (U30_rezaniNosniku.xlsx)

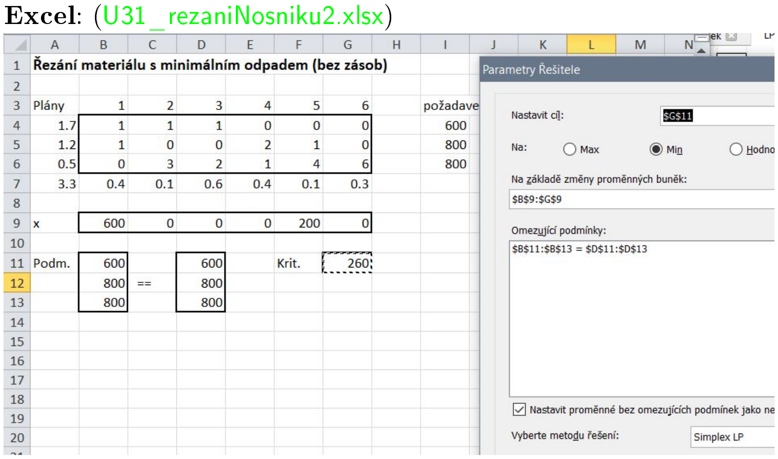


b) Splňte tytéž podmínky (z bodu a.) s minimální spotřebou materiálu, tj. neřežte do zásoby. Porovnejte výsledek s předchozím.

Řešení

Podmínky na rovnost.

Excel: (U31_rezaniNosniku2.xlsx)



c) Na jednu konstrukci jsou třeba dva kusy o délce 1,7 m, tři kusy délky 1,2 m a pět půlmetrových kusů. Jedna dodávka materiálu obsahuje 1000 kusů délky 3,3 m. Zajistěte rozřezání materiálu tak, aby bylo možno sestavit maximální počet konstrukcí.

Řešení

Stav x - počet řezání podle plánu, j - počet konstrukcí
Krit. - počet konstrukcí $\rightarrow$ max
Podm. - dodržení požadavků na jednotlivé délky

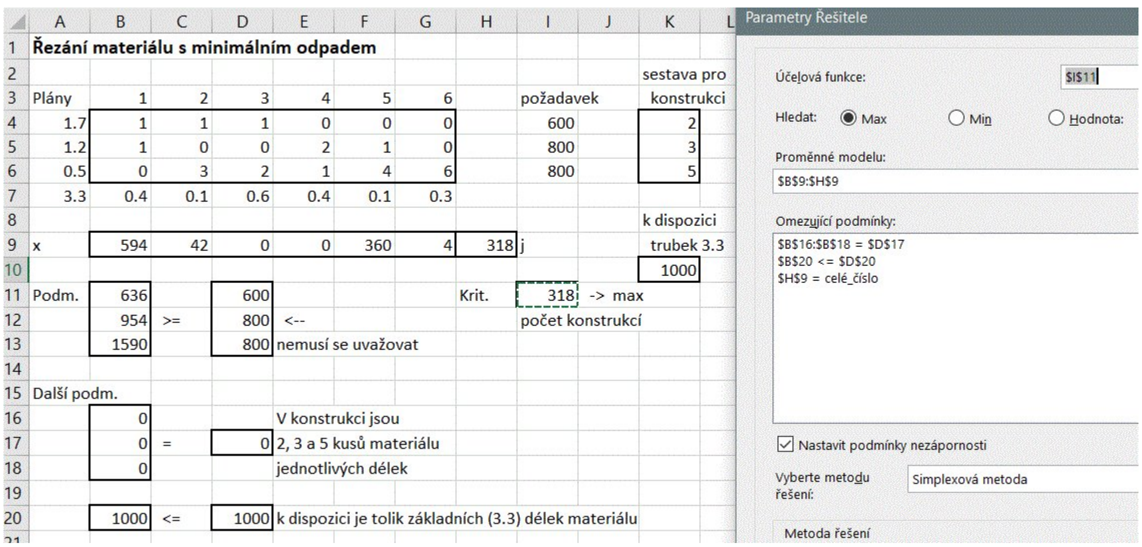
Další podmínka - vazba mezi počtem konstrukcí a počtem jednotlivých dílů. Zajímají nás jenom konstrukce, ne samostatné díly.

Omezení na počet základních trubek o délce 3.3 m.

Poznámka

Podmínky požadující minimální výrobu jednotlivých délek jsou tady trochu podivné. Nemusí se uvažovat. Jen pro menší počet 3.3 trubek způsobí, že úlohu nelze řešit.

Excel: (U32_rezaniNosniku3.xlsx)



	A	B	C	D	E	F	G	H	I	J	K
1	**Řezání materiálu s minimálním odpadem**										
2											sestava pro
3	Plány	1	2	3	4	5	6		požadavek		konstrukci
4	1.7	1	1	1	0	0	0		600		2
5	1.2	1	0	0	2	1	0		800		3
6	0.5	0	3	2	1	4	6		800		5
7	3.3	0.4	0.1	0.6	0.4	0.1	0.3				
8											k dispozici
9	x	594	42	0	0	360	4	318	j		trubek 3.3
10											1000
11	Podm.	636		600				Krit.	318	-> max	
12		954	>=	800	<--				počet konstrukcí		
13		1590		800	nemusí se uvažovat						
14											
15	Další podm.										
16		0			V konstrukci jsou						
17		0	=	0	2, 3 a 5 kusů materiálu						
18		0			jednotlivých délek						
19											
20		1000	<=	1000	k dispozici je tolik základních (3.3) délek materiálu						

2.11.2 Příklad

Řezání tyčí s možností sváření zbytků po řezání.

Z tyčí o délce 50 cm vyrábíme tyčky o délce 20 cm a 15 cm. Odpad lze dále svařovat na tyčky požadovaných délek (např. dva deseticentimetrové zbytky na jednu dvaceticentimetrovou tyčku atd.). Cena jedné původní tyčky je 3 koruny, cena jednoho „řezu" 2 Kč a cena jednoho sváru včetně zabroušení 0.1 Kč. Úkolem je navrhnout takový program řezání a svařování, aby podnik vyrobil alespoň 140 tyček délky 20 cm a alespoň 220 tyček o délce 15 cm tak, aby celkové náklady na jejich pořízení byly minimální.

Řešení

Řezné plány zvolíme (nebereme všechny možné ale jen vybrané).

Plán 1: řežeme 2 kusy 20cm. Počet řezů: 2. Zbytek 10cm.

Plán 2: řežeme 1 kus 20cm a 2 kusy 15cm. Zbytek 0cm.

Plán 3: řežeme 3 kusy 15cm. Počet řezů: 3. Zbytek 5cm.

Plán 4: svaříme zbytky ze dvou plánů 1. Počet svárů: 1. Dostaneme 1ks 20cm.

Plán 5: svaříme zbytky z plánu 1 a 3. Počet svárů: 1. Dostaneme 1 ks 15cm.

Plán 4: svaříme zbytky ze tří plánů 3. Počet svárů: 2. Dostaneme 1ks 15cm.

Řezné plány zapíšeme do tabulky (jde o počty dílů, které z plánu dostaneme)

	řezy			sváry		
plán	1	2	3	4	5	6
20cm	2	1	0	1	0	0
15cm	0	2	3	0	1	1
řezy/sváry	2	2	3	1	1	2
náklady/jed.	2	2	2	0.1	0.1	0.1
nákl. celk.$=c$	4	4	6	0.1	0.1	0.2

Stav $x = [x_1, x_2, x_3, x_4, x_5, x_6]$ zavedeme jako počty realizací jednotlivých plánů

Tabulku počtu řezů označíme T, a požadované počty tyček jednotlivých délek $b_i,\ i = 1, 2$.

Kriterium

$$J = \sum_j c_{i,j} x_j + 3 \sum_j x_j \to \min$$

kde 3 je cena tyčky.

Podmínky

1. požadavky na nařezané množství

$$\sum_j T_{i,j} x_j \geq b_i$$

2. dostatek odřezků na svařování (musí být z čeho svařovat)

– pro plán 4 potřebujeme dva plány 1

– pro plán 5 potřebujeme jeden plán a jeden plán 3

– pro plán 6 potřebujeme tři plány 3

tedy

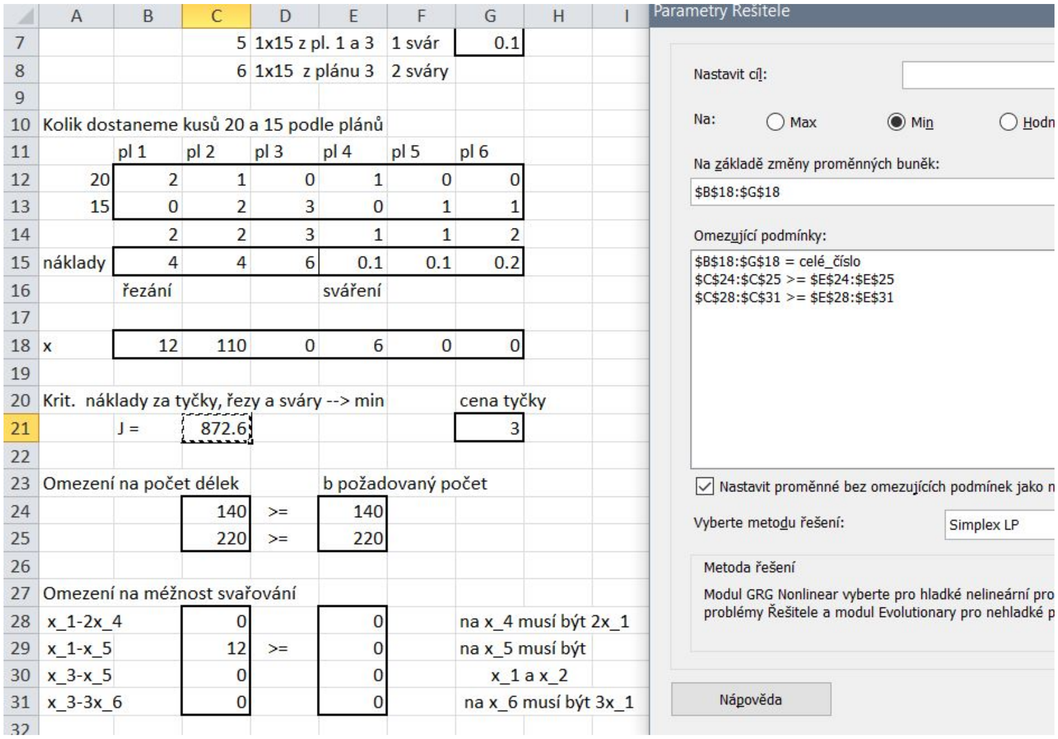
$$x_1 - 2x_4 \geq 0$$

$$x_1 - x_5 \geq 0$$

$$x_3 - x_5 \geq 0$$

$$x_3 - 3x_6 \geq 0$$

Excel: (U33_rezaniTyciSvarovani.xlsx)



	A	B	C	D	E	F	G	H	I
7			5	1x15 z pl. 1 a 3		1 svár	0.1		
8			6	1x15 z plánu 3		2 sváry			
9									
10	Kolik dostaneme kusů 20 a 15 podle plánů								
11		pl 1	pl 2	pl 3	pl 4	pl 5	pl 6		
12	20	2	1	0	1	0	0		
13	15	0	2	3	0	1	1		
14		2	2	3	1	1	2		
15	náklady	4	4	6	0.1	0.1	0.2		
16		řezání			sváření				
17									
18	x	12	110	0	6	0	0		
19									
20	Krit. náklady za tyčky, řezy a sváry --> min						cena tyčky		
21		J =	872.6				3		
22									
23	Omezení na počet délek				b požadovaný počet				
24			140	>=	140				
25			220	>=	220				
26									
27	Omezení na měžnost svařování								
28	x_1-2x_4		0		0		na x_4 musí být 2x_1		
29	x_1-x_5		12	>=	0		na x_5 musí být		
30	x_3-x_5		0		0		x_1 a x_2		
31	x_3-3x_6		0		0		na x_6 musí být 3x_1		
32									

Kapitola 3

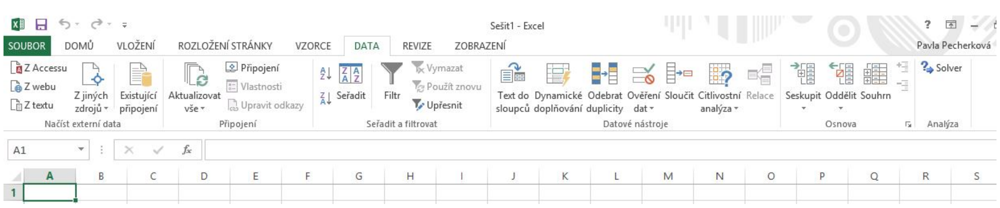
Programové vybavení

3.1 Excel

3.1.1 Řešitel

Pro řešení úloh lineárního programování je možné použít aplikaci Řešitel, tedy aplikaci, která slouží k vyhledávání extrémů funkce. Řešitel je součástí MS Excel. Výhoda této aplikace je v tom, že je přehledná a Excel je běžně využívaný a rozšířený tabulkový procesor. Nevýhoda je, že je omezený prostorem a pro složité úlohy se musí použít jiný program.

Všechny úlohy byly naprogramovány ve verzi MS Excel 2013 a i na tuto verzi optimalizovány. Z toho důvodu neručíme za nefunkčnost ve starších verzích. Tato verze je zdarma ke stažení na http://download.cvut.cz/. Řešitele naleznete po spuštění v záložce *DATA*, viz obrázek 3.1.1.

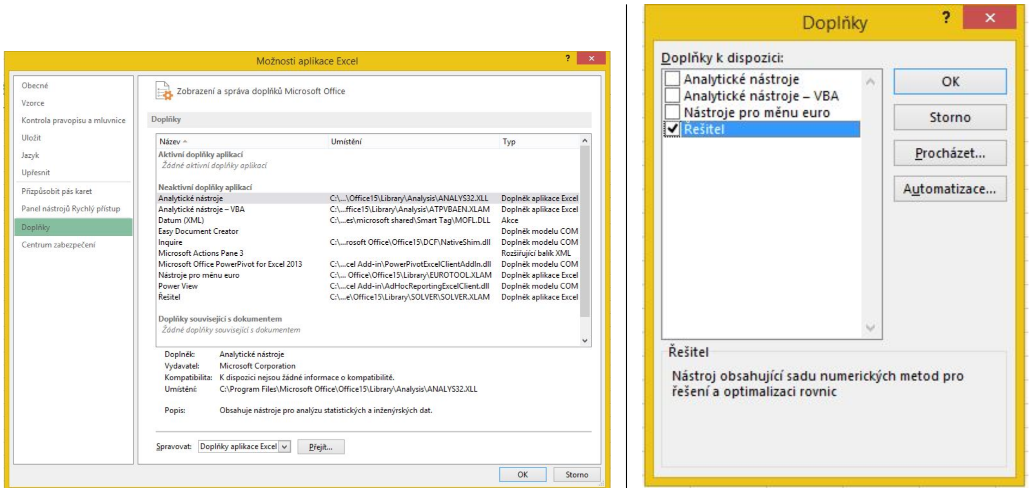


Obrázek 3.1.1: Excel - Data - Řešitel (Solver)

Pokud tam není, je nutné ho aktivovat. Lze aktivovat následujícím způsobem:

1. otevřete *"Soubor - Možnosti - Doplňky"* a otevře se vám nové okno, viz obrázek 3.1.2 (a),
2. v otevřeném okně dole klikneme na *"Spravovat: Doplňky aplikace Excel"* a otevře se další okno, viz obrázek 3.1.2 (b). Zaškrtneme *"Řešitel"* a potvrdíme pomocí tlačítka *"OK"*.

Pokud vše proběhlo v pořádku, v záložce *DATA* se objeví Řešitel (v anglické verzi Solver).



(a) (b)

Obrázek 3.1.2: Aktivace Řešitele

3.1.2 Model v sešitě Excel

1. Nejdříve vymezíme blok buněk pro neznámé x (případně další neznámé).
2. Dále zapíšeme všechny zadané veličiny - většinou to jsou ceny c a koeficienty lineárních omezení A a požadované pravé strany b.
3. Spočteme všechno, co je potřeba spočítat (do Řešitele se dávají jen odkazy na buňky nebo bloky buněk). Většinou to je:
   (a) kritérium $c'x = \sum_i c_i x_i$
   (b) vypočtené pravé strany omezení $Ax = \sum_j a_{ij} x_j \; \forall i$
4. Zavoláme Řešitele a zadáme všechny odkazy
   (a) zvolíme Min nebo Max pro kritérium,
   (b) zadáme blok nebo bloky pro neznámé (optimalizované) veličiny,
   (c) přidáme omezení ve tvaru $\leq$, $\geq$ nebo binární (volí se na stejném místě)
   (d) většinou necháme zatrženou volbu "neomezené veličiny jsou nezáporné" (nemusí se to již deklarovat v podmínkách)
   (e) a v okénku dole vybereme volbu "Simplex LP" - lineární programování.

3.1.3 Triky při práci v Excelu

Blok buněk - tažení myší nebo `Shift+šipka`.

Přemístění bloku (s `Ctrl` kopie) - uchopení za okraj (buňky nebo bloku) a tažení.

Vkopírování hodnoty do celého bloku - vytvoříme blok, zadáme hodnotu a `Ctrl+Enter`.

Maticové vzorce - provádějí operace (nejčastěji násobení) prvek po prvku. Zadávají se pomocí `Ctrl+Shift+Enter`. Výsledek může být v jedné buňce, nebo v bloku buněk. Blok je možno zadat předem (pak se objeví výsledky v celém bloku). Vzorec je možno kopírovat se všemi pravidly o posunu adres.

Fixace adres (absolutní adresy) - zafixované adresy se při kopírování neposouvají. Adresu fixuje dolar před písmenem, číslem nebo obojím. Dolar před písmenem fixuje adresu při vodorovném pohybu, před číslem při svislém pohybu a před obojím i ve vodorovném i svislém směru.

Zadání dolarů - automaticky zadáme dolary klávesou F4 ihned po vložení adresy (jinak se musí adresa dát do bloku). Opakované stisknutí F4 provede: oba dolary, jen před číslem, jen před písmenem, žádný dolar (atd.).

Skalární součin - pokud potřebujeme maticový výpočet $= \sum_i a_i b_i$ lze ho jednoduše vytvořit tak, že vytvoříme sumu součinu dvou sloupců a poté zmáčkneme `Ctrl+Shift+Enter`. Například `=suma(A1:A10*B1:B10) + Ctrl+Shift+Enter.`

Součin matice a vektoru - matice je `B1:D10` a vektor `A1:A10`. Součin je `=suma(B1:D10*$A$1:$A$10) + Ctrl+Shift+Enter` a rozkopírovat svisle. (Adresa s dolary je absolutní - viz fixace adres.)

3.1.4 Příklad v Excelu

Řešíme obecný příklad lineárního programování

$$5x_1 + x_2 \to \max$$

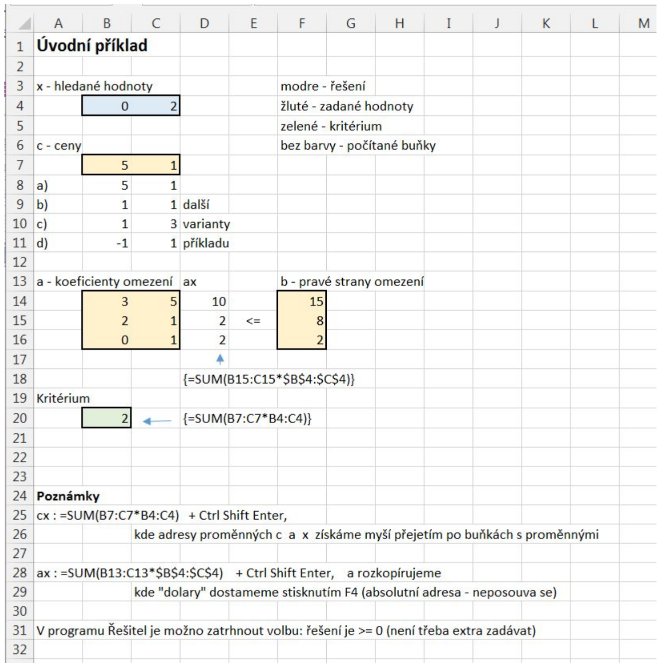
$$\begin{aligned} 3x_1 + 5x_2 &\leq 15 \\ 2x_1 + x_2 &\leq 8 \\ x_2 &\leq 2 \end{aligned}$$

$$x_1, x_2 \geq 0$$

Pro přehlednost, je hned na začátku uvedeno řešení, které je označeno modrým pozadím (obrázek 3.1.3). Dále jsou uvedeny váhy v kritériu (pro variantu (a) v obecném příkladu), tedy váhy $c = [5, 1]$. Následují podmínky jako matice

$$A = \begin{bmatrix} 3 & 5 \\ 2 & 1 \\ 0 & 1 \end{bmatrix}, \quad b = \begin{bmatrix} 15 \\ 8 \\ 2 \end{bmatrix}$$

které jsou také ve čtverečku s žlutým pozadím, stejně jako váhy v kritériu. Kritérium, které je označeno zeleným pozadím v rámečku, je základní parametr, kde se hledá maximum nebo minimum. Ani tento parametr neměníme. Při práci se obvykle mění pouze políčka s žlutým pozadím, tedy zadání.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	**Úvodní příklad**												
2													
3	x - hledané hodnoty					modre - řešení							
4		0	2			žluté - zadané hodnoty							
5						zelené - kritérium							
6	c - ceny					bez barvy - počítané buňky							
7		5	1										
8	a)	5	1										
9	b)	1	1	další									
10	c)	1	3	varianty									
11	d)	-1	1	příkladu									
12													
13	a - koeficienty omezení			ax		b - pravé strany omezení							
14		3	5	10		15							
15		2	1	2	<=	8							
16		0	1	2		2							
17													
18				{=SUM(B15:C15*$B$4:$C$4)}									
19	Kritérium												
20		2		{=SUM(B7:C7*B4:C4)}									
21													
22													
23													
24	**Poznámky**												
25	cx : =SUM(B7:C7*B4:C4) + Ctrl Shift Enter,												
26			kde adresy proměnných c a x získáme myší přejetím po buňkách s proměnnými										
27													
28	ax : =SUM(B13:C13*$B$4:$C$4) + Ctrl Shift Enter, a rozkopírujeme												
29			kde "dolary" dostameme stisknutím F4 (absolutní adresa - neposouva se)										
30													
31	V programu Řešitel je možno zatrhnout volbu: řešení je >= 0 (není třeba extra zadávat)												
32													

Obrázek 3.1.3: Příklad v Excelu

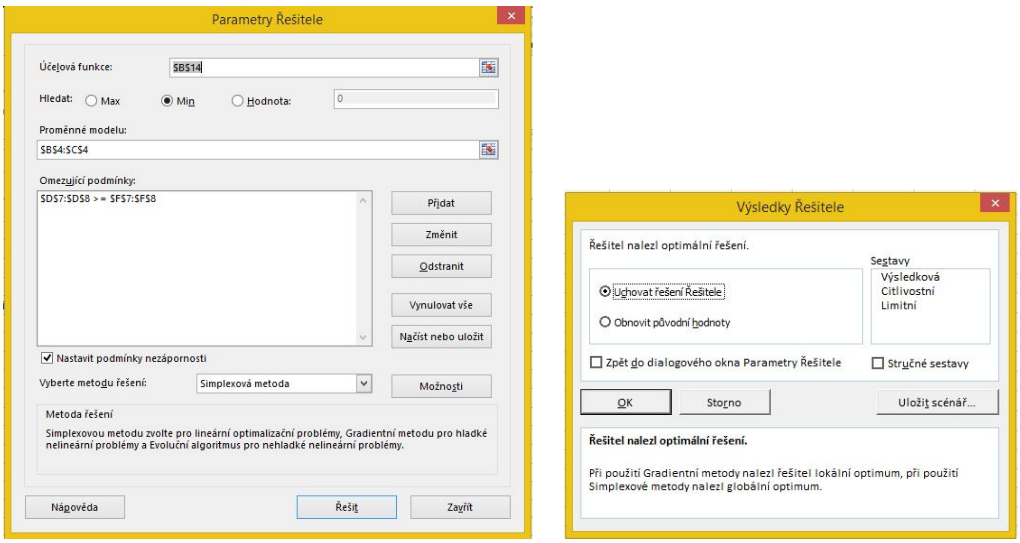
V případě, že již máme nastavené základní hodnoty, přecházíme k Řešiteli (Solver). Spustíme DATA-Řešitel (Solver) a objeví se nám tabulka znázorněna na obrázku 3.1.4 (a). Pro správnou funkci je potřeba nastavit následující:

1. **Účelová funkce** - odkaz na buňku s kritériem (zelené pozadí).
2. **Hledat** - hledá se minimum, maximum nebo určitá hodnota. Pro zadávajícího je důležité si uvědomit co hledá (u zisku bude hledat maximum a u nákladů minimum).
3. **Proměnné modelu** - výsledek (modré pozadí). Zde bude optimální hodnota pro zadané veličiny (například kolik kusů výrobku A a B vyrobit).
4. **Omezující podmínky** - zde musí být všechny podmínky, které je potřeba splnit, tzn. vytvoří se sloupcový vektor, do kterého se vypočítá součin výsledku (například kolik kusů vyrobit) s hodnotou řádku tak, aby se dal výsledek porovnat s omezením. Tedy „$a \cdot x$" porovnáme s pravou strannou omezení. Pokud bude ještě jiný požadavek, například minimální vyrobené množství, musí se zapsat také do Omezujících podmínek, kde se klikne na ikonku Přidat a nadefinuje se nová podmínka.
5. **Nastavit podmínky nezápornosti** - u většiny problémů se očekává, že výsledek nemůže být záporný (nemůžeme vyrobit -5 výrobků). Tato podmínka se může zadat přímo do omezujících podmínek nebo se zaškrtne políčko s podmínkou nezápornosti.
6. **Vyberte metou řešení** - na výběr je Gradientní metoda, Simplexová metoda a Evoluční algoritmus. Zvolení vhodné metody je na zadavateli a určí se podle typu úlohy.

(a) Simplexová metoda - lineární optimalizační úlohy,

(b) Gradientní metoda - hladké nelineární optimalizační úlohy,

(c) Evoluční algoritmus - nehladké nelineární optimalizační úlohy.



(a) (b)

Obrázek 3.1.4: Řešitel

Poté se již jen zadá funkce řešit a objeví se následující tabulka 3.1.4 (b). Tam necháme zaškrtnutou možnost **Uchovat řešení Řešitele** a potvrdit tlačítkem **OK**.

3.2 LiPS

LiPS - Linear Program Solver je free software, který provádí výpočet úloh lineárního (i celočíselného) programování. Navíc obsahuje také řešení úloh citlivosti a stability. Program není třeba instalovat, lze ho spustit přímo z .exe souboru, staženého na adrese

https://sourceforge.net/projects/lipside/

Tento software je doplňkový a budeme jej používat jednak jako ukázku simplexové metody, kterou ukazuje krok po kroku, jednak pro demonstraci citlivosti a stability úloh.

Poznámka

Podobný, poněkud silnější ale také složitější, je program LPSolve

https://sourceforge.net/projects/lpsolve/files/lpsolve/

Hodí se na rozměrné reálné úlohy.

Po spuštění programu LiPS se objeví aplikace s pracovní plochou. V ní je možno otevřít nové nebo již dříve uložené okno „model". To může mít dvojí podobu: *text* nebo *tabulka,* a to jak u nového, tak i u otevíraného okna (u otevíraného je typ okna možno vybrat v poli dole pod okénkem Název souboru).

Model se nejlépe zadá ve formě tabulky. Potom je možno model uložit a otevřít v textové podobě. Tím se zjistí, jaká jsou pravidla pro textové zadávání.

Řešení modelu se spustí ikonkou s bílým trojúhelníkem na zeleném poli. Po spuštění se objeví výsledkové okno „report", ve kterém je uložen celý postup řešení v simplexové tabulce. Výsledky jsou v posledních dvou tabulkách nadepsaných jako RESULTS.

První tabulka ukazuje řešení (Value), zadané ceny (Objective cost) a koeficienty stability (Reduced cost).

Druhá tabulka uvádí pravé strany omezení (RHS), rozdíly mezi levou a pravou stranou omezení (Slack) a koeficienty citlivosti (Dual price).

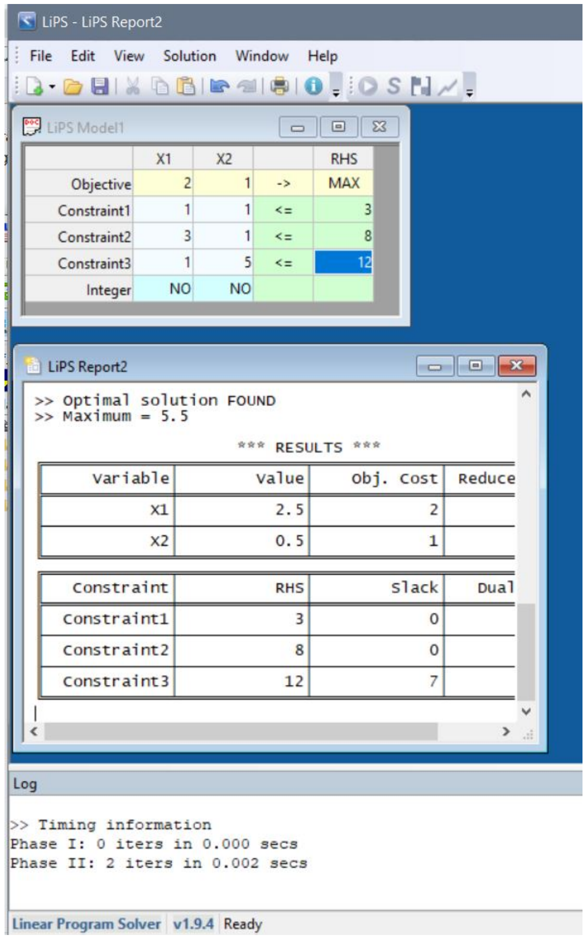
Po spuštění modelu a s kurzorem v okně modelu lze spustit další ikony vpravo od ikony Solve. Jsou to Sensitivity analysis, Matrix map a Solving history.

Po spuštění Sensitivity analysis (modré S) se objeví tabulka, kde vybereme RHS Range (All) a COST Range (All) a přesuneme je šipkou do pravého pole. Po OK se objeví citlivostní analýza.

Zbylé volby v předchozího menu provádí analýzu v situaci, kdy zvolíme jiné pravé strany nebo jiné koeficienty matice omezení A. Ty je třeba zadat tak, že se kurzorem postavíme na příslušný řádek v pravém okně a zvolíme dole Settings...

Zadaný model, stejně jako řešení nebo citlivost je možno uložit na disk.

Základní pohled na LiPS je v následujícím obrázku



3.3 Linear programming grapher

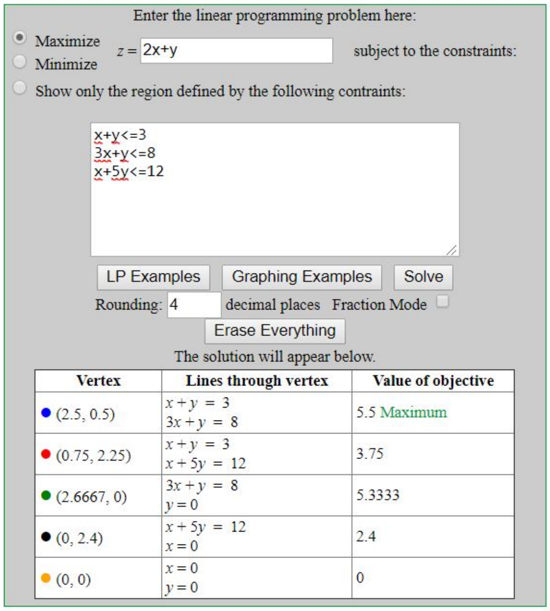
Jedná se o webovou aplikaci na adrese

https://www.zweigmedia.com/utilities/lpg/index.html?lang=en

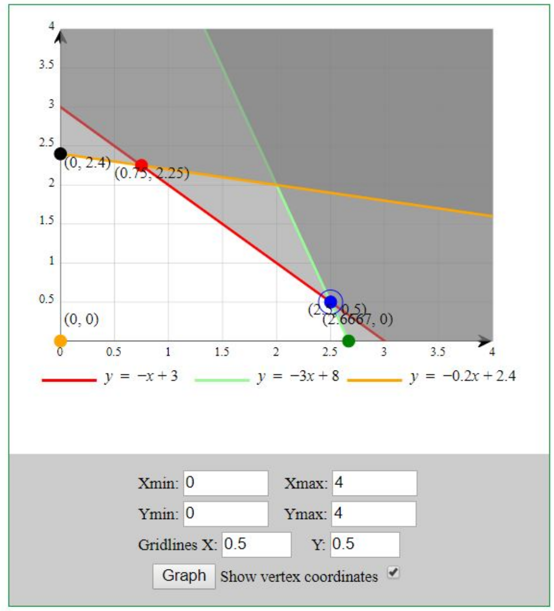
do které lze zadat model dvourozměrné úlohy lineárního programování a úlohu spustit tlačítkem Solve. Ukáže se přípustný simplex řešení s popisem jednotlivých hranic omezení a samozřejmě optimální řešení. Model se zadá nejlépe tak, že spustíte LP Examples a výsledek upravíte podle svého.

Vlevo nahoře, pod žlutou záložkou Main Page, je tlačítko Simplex method utility. Ta umožňuje řešit obecnou úlohu LP. Doporučuje se pracovat v nové verzi (červený text).

Pohled na Grapher je na následujícím obrázku



Vertex	Lines through vertex	Value of objective
(2.5, 0.5)	$x+y=3$, $3x+y=8$	5.5 Maximum
(0.75, 2.25)	$x+y=3$, $x+5y=12$	3.75
(2.6667, 0)	$3x+y=8$, $y=0$	5.3333
(0, 2.4)	$x+5y=12$, $x=0$	2.4
(0, 0)	$x=0$, $y=0$	0



Literatura

[1] P.Pecherková, I.Nagy: LP1Skripta - učební text (web)

[2] IBM: An Introduction to Linear Programming (web)

[3] D.G.Luenberger, Y.Ye: Linear and Nonlinear Programming, Springer, 2008, ISBN: 978-0-387-74502-2 (web)

[4] A. Kubišová: Operační výzkum. Vysoká škola polytechnická Jihlava, 2014 (web)

[5] J. Matoušek: Lineární programování. Úvod pro informatiky. KAM MFF UK, 2006 (web)