Návod na použití řešitele MS Excel a MPL for Windows

Jan Fábry

Obsah

1 Úloha výrobního plánování

1.1 Matematický model

Příklad:

Firma vyrábí dva typy dřevěných hraček: nákladní autíčka a vláčky. Autíčko prodává za 550 Kč, vláček za 700 Kč. Náklady na dřevo, které se při výrobě používá jako hlavní materiál, činí pro autíčko 50 Kč, pro vláček 70 Kč. Na výrobě obou hraček se podílejí řezbáři a lakýrníci, přičemž na jedno autíčko je zapotřebí 1 hodina řezbářské práce a 1 hodina dokončovací práce, na jeden vláček 2 hodiny řezbářské práce a 1 hodina dokončovací práce. Náklady na řezbářskou práci činí 30 Kč/hod, na dokončovací práci 20 Kč/hod. Každý měsíc je k dispozici 5000 hodin řezbářské práce a 3000 hodin dokončovací práce. Vzhledem k odbytu firma smí vyrobit maximálně 2000 autíček za měsíc, výroba vláčků není z hlediska poptávky omezena. Cílem firmy je najít výrobní program, který bude maximalizovat zisk, vyjádřený jako rozdíl tržeb a nákladů.

Jedná se o typickou **úlohu výrobního plánování**. Ve firmě lze identifikovat dva procesy: výrobu autíček a výrobu vláčků. Cílem firmy je maximalizace celkového zisku. Na výrobě hraček se podílejí dva činitelé: řezbářská práce a dokončovací práce. Dalším činitelem, ovlivňujícím výrobní proces, je poptávka po autíčkách.

Proměnné

V lineárním matematickém modelu vystupují dvě proměnné:

x_1 = počet autíček vyrobených za měsíc,

x_2 = počet vláčků vyrobených za měsíc.

Přesná definice všech proměnných je nezbytná především pro závěrečnou interpretaci získaných výsledků. Z tohoto hlediska je důležité stanovit i časové období, k němuž se uvedené proměnné vztahují (v tomto případě jeden měsíc).

Účelová funkce

Jakmile jsou známy všechny proměnné, lze přistoupit k formulaci účelové funkce, která v tomto případě představuje závislost hodnoty celkového zisku na výši produkce. Celkový měsíční zisk lze vyjádřit jako rozdíl celkových měsíčních tržeb, plynoucích z prodeje vyrobených výrobků, a celkových měsíčních nákladů na produkci:

$$\textit{zisk} = \textit{tržby} - \textit{náklady}. \qquad (1)$$

Při popisu problému jsme předpokládali, že firma veškeré vyrobené hračky také prodá (za stanovenou cenu). Proto lze funkci celkových měsíčních tržeb vyjádřit následovně:

$$\textit{tržby} = 550x_1 + 700x_2 \qquad (2)$$

Tržby za každý typ produktu jsou součinem jeho ceny a vyrobeného (prodaného) množství.

V kalkulaci celkových měsíčních nákladů musíme vzít v úvahu, které nákladové položky jsou v problému zahrnuty. Jedná se o náklady na použité dřevo, náklady na řezbářskou práci a náklady na dokončovací práci. Nejprve pro každý typ produktu vypočteme jednotkové náklady, přičemž u řezbářské a dokončovací práce je k tomuto účelu nutné vynásobit cenu práce počtem hodin, nezbytných pro výrobu 1 ks produktu:

jednotkové náklady na autíčko = 50 + 30.1 + 20.1 = 100 Kč, (3)

jednotkové náklady na vláček = 70 + 30.2 + 20.1 = 150 Kč. (4)

Funkce celkových měsíčních nákladů je

$$náklady = 100x_1 + 150x_2 \,. \tag{5}$$

Po dosazení funkcí (2) a (5) do (1) získáváme funkci celkového měsíčního zisku:

$$zisk = 450x_1 + 550x_2 \,. \tag{6}$$

Omezující podmínky

Při výrobě autíček a vláčků je použita technologie, která já náročná na dva druhy práce, a sice řezbářskou a dokončovací. V typických příkladech z oblasti výrobního plánování často vystupují v roli významného činitele suroviny či materiál s omezenou zásobou. I v našem příkladu je dřevo základním materiálem, nebereme jej však v úvahu jako činitele ovlivňujícího výrobní proces[1]. Zásoba dřeva (u dodavatele firmy) je zřejmě natolik velká, že v podmínkách malé firmy nelze předpokládat její úplné vyčerpání.

Nejprve sestavíme omezující podmínku, která zaručuje, že nebude překročen počet hodin vymezený pro řezbářskou práci:

$$x_1 + 2x_2 \le 5000 \,. \tag{7}$$

Levá strana nerovnosti představuje skutečnou spotřebu řezbářské práce (v hodinách) za jeden měsíc výroby, hodnota pravé strany vyjadřuje disponibilní počet hodin řezbářské práce v jednom měsíci, uvedený v zadání úlohy.

Podobně formulujeme podmínku, týkající se dokončovací práce:

$$x_1 + x_2 \le 3000 \,. \tag{8}$$

Posledním činitelem ovlivňujícím výrobu hraček je poptávka po autíčkách. Výroba tohoto typu hraček nesmí překročit zadaný limit 2000 kusů, což lze vyjádřit následující nerovností:

$$x_1 \le 2000 \,. \tag{9}$$

Omezující podmínky, zachycující vliv činitelů, se označují jako *vlastní omezení* úlohy LP. Kromě těchto podmínek musí proměnné splňovat ještě další omezení, tzv. *podmínky nezápornosti*:

$$x_1 \ge 0, x_2 \ge 0 \,. \tag{10}$$

Protože proměnné představují počty vyrobených hraček, musí nabývat jen celočíselných hodnot. Posledními podmínkami budou tedy tzv. *podmínky celočíselnosti*:

$$x_1, x_2 \text{ – celé.} \tag{11}$$

[1] Spotřeba dřeva se v matematickém modelu objevuje jen ve spojitosti s náklady.

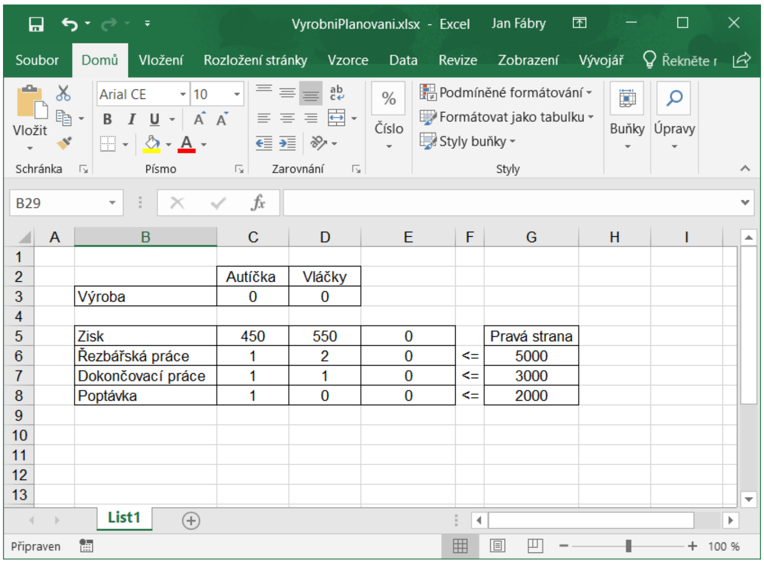
Úplná formulace matematického modelu je

$$
\begin{aligned}
& z = 450x_1 + 550x_2 \rightarrow \max, \\
& x_1 + 2x_2 \le 5000, \\
& x_1 + x_2 \le 3000, \\
& x_1 \le 2000, \\
& x_1, x_2 \ge 0, \\
& x_1, x_2 - \text{ celé}.
\end{aligned}
\tag{12}
$$

1.2 Řešení úlohy v MS Excel

Prostředí MS Excel nabízí uživateli možnost řešit jednodušší úlohy LP pomocí tzv. **Řešitele**, který je v omezené verzi součástí instalace MS Office[2]. Jeho použití budeme demonstrovat na výše uvedeném příkladu.

1. Nejdříve zadáme do tabulky koeficienty, obsažené v matematickém modelu (viz Obr. 1). Údaje ve sloupci C se týkají autíček, sloupec D odpovídá vláčkům. Koeficienty účelové funkce zadáme do buněk v oblasti C5:D5, koeficienty u proměnných v omezujících podmínkách do oblasti C6:D8 a hodnoty pravých stran do buněk G6:G8. V tabulce je nutné vymezit buňky pro hodnoty proměnných x_1 a x_2. Na Obr. se jedná o oblast C3:D3. Na počátku můžeme do těchto buněk vložit nuly.

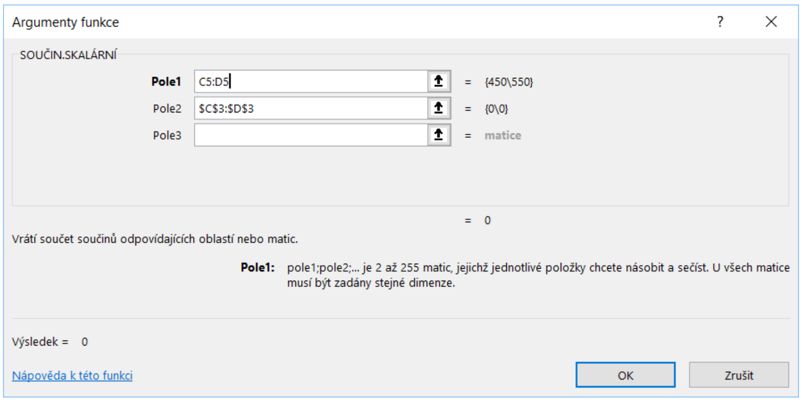


Obr. 1 Zadání úlohy LP v tabulce MS Excel

2. K zápisu matematického modelu v prostředí MS Excel je zapotřebí vložit do buněk výrazy, představující účelovou funkci a levé strany omezujících podmínek. Pro tento účel jsme v tabulce na Obr. 1 vyhradili sloupec E. Účelová funkce je z matematického hlediska skalárním součinem vektoru koeficientů účelové funkce (oblast C5:D5) a vektoru proměnných (oblast C3:D3). Využijeme-li pro vložení účelové funkce do buňky E5 průvodce zadávání funkcí, najdeme v seznamu funkci SOUČIN.SKALÁRNÍ a výše uvedené oblasti označíme podle Obr. 2. Jak se ukáže později, bude účelné adresy buněk, obsahujících

[2] Řešení je provedeno ve verzi MS Office 365 ProPlus.

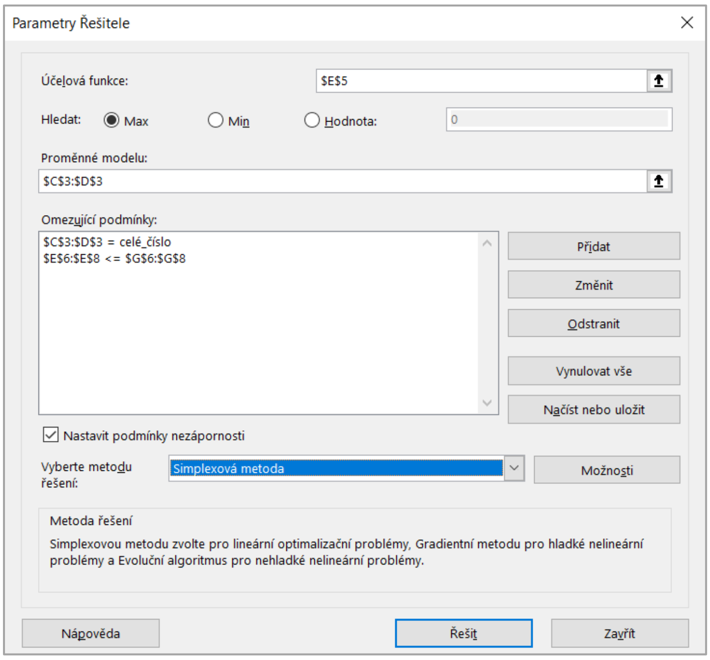
proměnné tzv. „zafixovat", tj. zadat je se znaky $ (dolar) jako absolutní adresy[3]. Protože hodnoty proměnných jsou nulové, objeví se v buňce E5 také nula.



Obr. 2 Využití funkce SOUČIN.SKALÁRNÍ pro zadání účelové funkce

Výrazy, odpovídající levým stranám omezujících podmínek (buňky v oblasti E6:E8), jsou skalárními součiny vektoru koeficientů v příslušném řádku a vektoru proměnných. Protože jsme při zadávání účelové funkce použili absolutní adresy pro oblast proměnných a relativní adresy pro oblast koeficientů účelové funkce, stačí buňku E5 zkopírovat do buněk v oblasti E6:E8. Do sloupce F jsme z důvodu přehlednosti zapsali typ omezujících podmínek (<=).

3. K úplnému zadání matematického modelu a vyřešení úlohy v MS Excel je určen nástroj *Řešitel*, který se nachází na kartě *Data* ve skupině *Analýza*. Pokud se na kartě tento nástroj nenachází, je nutné jej zpřístupnit jako doplněk. V rámci volby *Soubor* vybereme *Možnosti* a dále zvolíme *Doplňky*. V dolní části okna klikneme na tlačítko *Přejít…* a v následujícím okně *Doplňků* zaškrtneme nástroj *Řešitel*. Tímto jednoduchým postupem zpřístupníme *Řešitele* na kartě *Data* a můžeme přistoupit k zadání matematického modelu v dialogovém okně *Parametry Řešitele*. Na Obr. 3 je již finální podoba vyplněného okna.



Obr. 3 Zadání matematického modelu v *Řešiteli*

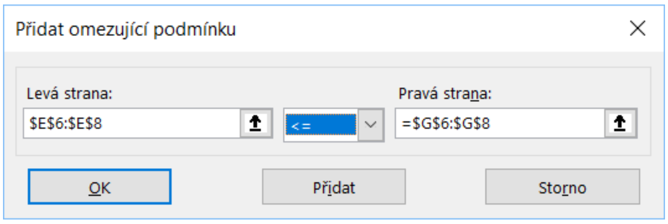
[3] K tomuto účelu je velice efektivní použít možnosti cyklické změny absolutního a relativního adresování buněk pomocí funkční klávesy F4, kterou stiskneme hned po výběru příslušné oblasti buněk.

Jednotlivé kroky nyní podrobně popíšeme:

a) Do dialogového rámečku *Účelová funkce* je nutné vybrat buňku, v níž je zadána účelová funkce. V našem případě se jedná o buňku E5 se zadanou funkcí SOUČIN.SKALÁRNÍ. Zároveň je nutné pomocí přepínače vybrat typ extrému, který v úloze hledáme, v naší úloze *Max*. Možnost *Hodnota* lze vybrat v případě, že máme představu o hodnotách účelové funkce a hledáme takové řešení, které této hodnoty dosahuje, tedy nemusí jít nutně o nejlepší řešení.

b) V dialogovém rámečku *Proměnné modelu* je nutné specifikovat adresy oblastí, které jsou vyhrazeny pro všechny proměnné v modelu. V případě, že se jedná o několik nesouvislých oblastí, je nutné při jejich výběru myší použít klávesy CTRL. Jednotlivé oblasti jsou pak v rámečku odděleny středníkem.

c) Nejsložitějším krokem je zadání omezujících podmínek. Po kliknutí na tlačítko *Přidat* se otevře další dialogové okno, do kterého je zapotřebí vybrat buňku, v níž je zadána *Levá strana* omezující podmínky, dále typ omezení a nakonec buňku, v níž je *Pravá strana*. Pokud jsou v tabulce zadány omezující podmínky *stejného typu* v souvislé oblasti jako v našem případě (typ <=), lze zadat všechna omezení v rámci jednoho dialogového okna, jak je zřejmé z Obr. 4. K tomuto účelu je užitečné seřadit všechny omezující podmínky tak, aby byly rozděleny do tří bloků podle typu podmínek. V lineárním programování neexistuje možnost zadání omezujících podmínek ve tvaru ostrých nerovností „<“ a „>“. Kromě tří možností „<=“, “=“ a „>=“ lze v rozbalovacím seznamu najít i další dvě možnosti „celé“ a „binární“ pro zadání celočíselných proměnných a tzv. binárních proměnných (viz dále), a také speciální typ „různé“, který ovšem není standardní podmínkou úlohy LP. Slouží pro vyjádření podmínky, že všechny hodnoty ve specifikovaných buňkách mají být vzájemně odlišné.



Obr. 4 Zadání omezujících podmínek

V tomto okně můžeme hned zadat také podmínky celočíselnosti pro obě proměnné. Po kliknutí na tlačítko *Přidat* do dialogového rámečku *Levá strana* vybereme buňky v oblasti C3:D3, tedy buňky, které jsou vyhrazeny pro proměnné modelu. Jak bylo uvedeno, v tomto případě je typ omezující podmínky nastaven na „celé“. Kliknutím na tlačítko *OK* se vrátíme do hlavního okna *Parametry Řešitele*.

d) Podmínky nezápornosti jsou defaultně přednastaveny pod oknem *Omezující podmínky* zaškrtnutím políčka *Nastavit podmínky nezápornosti*.

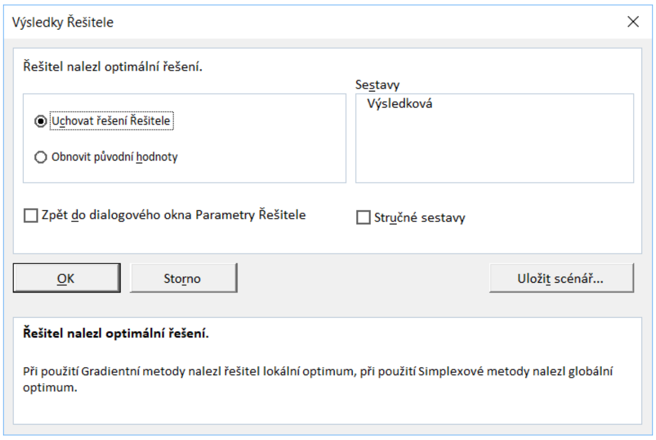
e) Velice důležitým krokem je výběr metody. V příslušném rozbalovacím seznamu vybereme *Simplexovou metodu*, která je standardní metodou pro řešení úloh LP[4].

f) Jestliže úloha obsahuje celočíselné či binární proměnné, je nutné se ještě věnovat nastavení přesnosti vypočtených hodnot po klinutí na tlačítko *Možnosti*. Hodnota

[4] Toto označení není zcela přesné, neboť pro úlohy LP, v nichž se vyskytují celočíselné nebo binární proměnné (viz dále) se používají speciální metody, nikoli samotná simplexová metoda. Na druhé straně je však nutné připustit, že se ve zmíněných metodách simplexová metoda v určité podobě tak jako tak objevuje.

0,000001 v dialogovém rámečku *Přesnost omezující podmínky* představuje toleranci pro rovnost hodnoty pravé a levé strany omezujících podmínek. Vysvětlení je poměrně jednoduché. Pokud v průběhu výpočtu např. podíl 1/3 vynásobíme třemi, není možné v paměti počítače získat matematicky přesnou hodnotu 1, ale např. 0,99999999999. Takové číslo, vzniklé jako důsledek zaokrouhlovacích chyb, při nastavení tolerance, se kterou daný program pracuje, lze ovšem za hodnotu 1 považovat. Naopak číslo 0,9999 v žádném případě nepředstavuje hodnotu 1, proto musí být tolerance nastavena tak, aby tato záměna nenastala. Podobný princip se týká i hodnot celočíselných či binárních proměnných. Jedná se o hodnotu v rámečku *Optimalita celých čísel (%)*. Přednastavená hodnota je 1, je nutné ji změnit např. také na 0,000001. Na tomto místě je vzhledem ke vzniku zaokrouhlovacích chyb vhodné podotknout, že před každým spuštěním výpočtu je zapotřebí vynulovat všechny buňky, v nichž jsou definovány proměnné. Řešitel totiž vždy vychází právě z hodnot, které jsou momentálně v těchto buňkách a při opakovaném spuštění může docházet ke kumulaci chyb, způsobených zaokrouhlováním.

4. Nyní přistoupíme k samotnému řešení úlohy a interpretaci výsledků. Výpočet spustíme kliknutím na tlačítko *Řešit* v hlavním oknu *Parametry Řešitele*. Po dokončení výpočtu se objeví dialogové okno *Výsledky Řešitele* se základní informací, zda bylo nalezeno požadované řešení (viz Obr. 5). V našem případě *Řešitel nalezl optimální řešení*.



Obr. 5 Dialogové okno *Výsledky Řešitele*

Dále je nutné nastavit dvě základní možnosti, týkající se výstupů:

a) V levé části okna lze vybrat, zda má být v tabulce Excelu uchováno právě získané řešení (tedy hodnoty proměnných a všechny hodnoty vypočtené v příslušných buňkách) nebo zda se mají obnovit původní hodnoty, tedy hodnoty, které se v buňkách proměnných nacházely před spuštěním výpočtu (vzhledem k výše uvedenému problému se zaokrouhlováním by to měly být vždy nulové hodnoty). Doporučuje se vždy vybrat spíše první možnost.

b) Pravá část okna obsahuje seznam sestav, které nabízí *Řešitel*. Pokud se v modelu nevyskytují celočíselné, ani binární proměnné, obsahuje seznam tři typy sestav: *Výsledkovou*, *Citlivostní* a *Limitní*. V opačném případě získáme pouze *Výsledkovou* sestavu. Pokud daný typ sestavy požadujeme vytvořit, je nutné jej vybrat kliknutím myši.

Potvrzením výběru kliknutím na tlačítko *OK* se vrátíme přímo do sešitu Excelu, do něhož byl přidán list *Výsledková sestava 1* (číslo udává pořadí vytvořené sestavy v případě, že řešení úlohy spouštíme opakovaně např. při jiném nastavení parametrů). Zároveň si lze všimnout, že v buňkách proměnných C3 a D3 skutečně zůstaly vypočtené hodnoty, a to 1000 a 2000 (viz Obr. 6). Samozřejmě se změnily i hodnoty ve všech buňkách obsahujících příslušné vzorce.

	A	B	C	D	E	F	G	H
1								
2			Autíčka	Vláčky				
3		Výroba	1000	2000				
4								
5		Zisk	450	550	1550000		Pravá strana	
6		Řezbářská práce	1	2	5000	<=	5000	
7		Dokončovací práce	1	1	3000	<=	3000	
8		Poptávka	1	0	1000	<=	2000	
9								

Obr. 6 Výsledek optimalizace přímo v listu sešitu

Vzhledem k tomu, že získané řešení již známe z předešlé podkapitoly, v níž jsme řešili úlohu graficky, není nutné se znovu interpretovat všechny hodnoty. Hodnota účelové funkce je v buňce E5, hodnoty levých stran omezujících podmínek jsou, jak již bylo dříve uvedeno, v oblasti E6:E8. Je na první pohled evidentní, že všechny tři omezující podmínky jsou splněny, první dvě jsou splněny jako rovnice. V horní části listu *Výsledková sestava 1* jsou uvedeny základní informace o výpočtu, včetně doby jeho trvání, vybrané metody řešení, výše nastavených tolerancí apod. V další části sestavy pak nalezneme strukturované výsledky (viz Obr. 7).

14	Účelová funkce (Max)						
15	**Levá strana omezující podmínky**	**Název**	**Původní hodnota**	**Konečná hodnota**			
16	$E$5	Zisk	0	1550000			
17							
18							
19	Proměnné						
20	**Levá strana omezující podmínky**	**Název**	**Původní hodnota**	**Konečná hodnota**	**Celé_číslo**		
21	$C$3	Výroba Autíčka	0	1000	Celé_číslo		
22	$D$3	Výroba Vláčky	0	2000	Celé_číslo		
23							
24							
25	Omezující podmínky						
26	**Levá strana omezující podmínky**	**Název**	**Hodnota**	**Vzorec**	**Stav**	**Odchylka**	
27	$E$6	Řezbářská práce	5000	$E$6<=$G$6	Platí	0	
28	$E$7	Dokončovací práce	3000	$E$7<=$G$7	Platí	0	
29	$E$8	Poptávka	1000	$E$8<=$G$8	Neplatí	1000	
30	$C$3:$D$3=Celé_číslo						
31							

Obr. 7 Výsledky v listu *Výsledková sestava 1*

Nejprve je uvedena tabulka týkající se *Účelové funkce*. *Původní hodnota* se týká obsahu buňky účelové funkce před spuštěním optimalizace, *Konečná hodnota* pak obsahu této buňky po jejím dokončení. V tabulce nazvané *Proměnné* lze nalézt podobné údaje o buňkách obsahujících proměnné modelu. V posledním sloupci je navíc uvedena informace o tom, že proměnné jsou celočíselné. V poslední tabulce *Omezující podmínky* ve sloupci *Hodnota* jsou vypočtené hodnoty levých stran omezujících podmínek pro optimální řešení. V případě, že se pro danou nerovnici ve výsledku hodnota levé strany rovná hodnotě pravé strany, je ve sloupci *Stav* uvedeno *Platí*. V opačném případě je uvedeno *Neplatí*. S tím souvisí i poslední sloupec, který obsahuje hodnotu *Odchylky* jako rozdíl levé a pravé strany příslušné omezující podmínky. Jedná se o hodnoty již dříve zmíněných přídatných proměnných, které budou podrobněji definovány a vysvětleny později.

Omezující podmínky ve tvaru rovnice mají v příslušném řádku samozřejmě uvedeno *Platí* a hodnota odchylky je rovna 0. *Výsledková sestava* má svůj význam při komunikaci analytika se zadavatelem problému, kterému lze tento list poskytnout s interpretací výsledků, případně spolu s dalšími informacemi o průběhu, které ho mohou zajímat (např. délka samotného výpočtu, existence alternativního optimálního řešení apod.)[5].

MS Excel je určen pro řešení jednodušších a méně rozsáhlých úloh lineárního programování. Reálné úlohy ovšem obsahují desítky, stovky i tisíce proměnných a podobný počet omezujících podmínek. V takovém případě nelze *Řešitele* využít. To platí bohužel i pro některé typy úloh, které mají sice menší rozsah, ale obsahují celočíselné či binární proměnné (viz dále). Řešení takových úloh naráží na problém s jejich výpočetní složitostí a použití takového jednoduchého nástroje, jakým je bezesporu *Řešitel* MS Excel, je prakticky vyloučeno.

1.3 Řešení úlohy v MPL for Windows

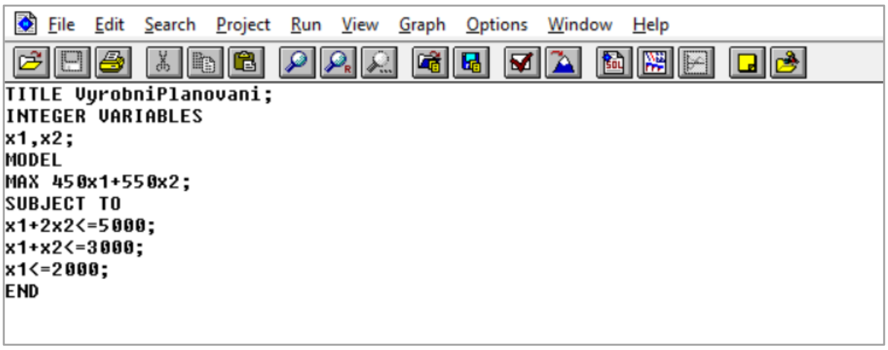
V této části se seznámíme s profesionálním přístupem k řešení úloh lineárního programování pomocí specializovaného softwaru. Profesionální specializované optimalizační produkty lze rozdělit na dvě základní skupiny:

- systémy na podporu modelování,
- optimalizační systémy (řešitelé).

První skupinu tvoří programy, které analytik využívá k počítačovému zápisu matematického modelu v příslušné syntaxi. Druhou skupinu tvoří optimalizační nástroje, které řeší danou úlohu. Pro lepší pochopení rozdílu mezi oběma typy programů lze uvést analogii s řešením úlohy LP v prostředí MS Excel. Pro zapsání matematického modelu byl využitý samotný MS Excel, tedy prostředí tabulkového kalkulátoru ve smyslu zadání jednotlivých částí modelu ve formě konstant a funkcí, které jsou vloženy v jednotlivých buňkách. K formulaci modelu bylo dále využito dialogového okna *Parametry Řešitele*. Lze si tedy představit, že systému na podporu modelování se věnují první tři kroky postupu uvedeného v předchozí podkapitole. Optimalizační systém, tedy *Řešitel*, je aktivován až ve čtvrtém kroku, kdy je spuštěn samotný výpočet.

Systémy na podporu modelování lze rozdělit na uzavřené a otevřené. Zatímco uzavřené systémy mají svého vlastního řešitele a nelze k řešení využít žádný jiný nástroj (to je právě případ MS Excel a jeho *Řešitele*), otevřené systémy umožňují uživateli výběr z několika řešitelů. Je pak na něm, pro který z nich se rozhodne.

V této části bude popsán postup zadání matematického modelu v profesionálním systému MPL for Windows (dále jen MPL) od firmy Maximal Software[6] a jeho řešení pomocí řešitele CPLEX od firmy IBM. Na Obr. 8 je v editoru MPL zadán matematický model ve své nejjednodušší podobě v editoru MPL.



Obr. 8 Zadání modelu v editoru MPL for Windows

[5] Později bude podrobněji popsána i další výstupní zpráva nazvaná *Limitní sestava*.
[6] http://www.maximalsoftware.com/

Jedná se o stejnou úlohu, která byla řešena v *Řešiteli* MS Excel. Později ukážeme na složitější úloze LP profesionální práci s tímto modelovacím systémem.

V tomto jednoduchém zápisu modelu jsou významné následující sekce a klíčová slova:

TITLE
Klíčové slovo pro definici názvu modelu.

INTEGER VARIABLES
V této části se definují celočíselné proměnné. V případě binárních proměnných se místo slova INTEGER použije slovo BINARY. Pokud mají proměnné splňovat pouze podmínky nezápornosti, bude uvedeno pouze VARIABLES. V modelu mohou být definovány samozřejmě všechny tři typy proměnných. Proměnné se zapisují bez použití horních a dolních indexů.

MODEL
Toto klíčové slovo označuje, že bude následovat definice modelu.

MAX
Tato zkratka se používá pro maximalizační úlohu, pro minimalizační úlohu je logicky vyhrazena zkratka MIN. Následuje matematický zápis účelové funkce.

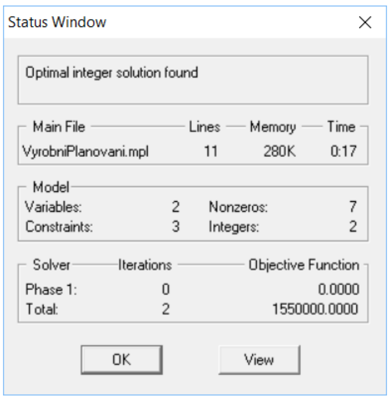
SUBJECT TO
Začátek sekce vyhrazené pro omezující podmínky, které se zadávají způsobem zřejmým z obrázku.

END
Toto slovo je posledním slovem celého zápisu.

Je nutné dodržet psaní středníků na konci každého řádku (kromě řádků s názvy sekcí a klíčovými slovy). Ve složitějších modelech je vhodné používat vysvětlující poznámky, které jsou uvozené vykřičníkem.

Před spuštěním výpočtu lze vybrat řešitele, což samozřejmě předpokládá, že je daný řešitel nainstalován. ŠAVŠ používá plnou akademickou verzi MPL spolu s plnou verzí řešitele CPLEX. Samotné spuštění optimalizace lze provést výběrem položky Solve Cplex v menu Run nebo kliknutím na následující ikonu: .

Po dokončení výpočtu se otevře okno s informací o tom, jakého výsledku řešitel dosáhl (viz Obr. 9).



Obr. 9 Výsledek optimalizace v MPL

I v tomto případě bylo samozřejmě nalezeno optimální celočíselné řešení (*Optimal integer solution found*). V okně jsou další užitečné informace jako je doba výpočtu (*Time*), počet všech proměnných v modelu (*Variables*), počet celočíselných proměnných (*Integers*), počet omezujících podmínek (*Constraints*), počet nenulových koeficientů u proměnných (*Nonzeros*), a především hodnota účelové funkce (*Objective Function*). Kliknutím na tlačítko *View* otevřeme soubor s výsledkovou zprávou (viz Obr. 10). Opět není nutné interpretovat již známé výsledky, je zapotřebí jen okomentovat některé speciality prostředí MPL. Především je třeba zdůraznit, že hodnoty ve sloupcích *Reduced Cost* a *Shadow Price* nemají v případě celočíselné úlohy žádný význam a budou interpretovány později v úlohách bez celočíselných a binárních proměnných. Je evidentní, že sloupec nazvaný *Slack* je analogií sloupce *Odchylka* ve výsledkové sestavě *Řešitele* MS Excel. Na rozdíl od něj ovšem hodnoty neudávají rozdíl mezi levou a pravou stranou nerovnice v absolutní hodnotě, ale jsou kladné v případě omezení typu $\leq$ a záporné v případě $\geq$. K tomuto označení se ještě vrátíme později při definici přídatných proměnných.

```
SOLUTION RESULT

  Optimal integer solution found

    MAX Obj              =     1550000.0000

DECISION VARIABLES

PLAIN VARIABLES

    Variable Name              Activity     Reduced Cost
  ------------------------------------------------------
    x1                       1000.0000         450.0000
    x2                       2000.0000         550.0000
  ------------------------------------------------------


CONSTRAINTS

PLAIN CONSTRAINTS

    Constraint Name               Slack     Shadow Price
  ------------------------------------------------------
    c1                           0.0000           0.0000
    c2                           0.0000           0.0000
    c3                        1000.0000           0.0000
  ------------------------------------------------------

END
```

Obr. 10 Výsledková sestava MPL

Pokud okno s výsledkovou zprávou omylem zavřeme, není nutné znovu spouštět výpočet. Stačí kliknout na následující ikonu:  .

2 Dopravní problém

Dopravní problém (DP) je speciální úlohou LP, v níž množina ekonomických subjektů, označovaných jako *dodavatelé*, nabízí určitý homogenní produkt, surovinu, materiál apod. jiným subjektům, označovaným jako *odběratelé*. Každý dodavatel je charakterizován *kapacitou*, tj. množstvím jednotek, které je schopen dodat. Každý odběratel potřebuje získat určité množství jednotek, které budeme nazývat jeho *požadavkem*.

V reálné praxi se při přepravě mezi jednotlivými dodavateli a odběrateli velice často uvažují *jednotkové přepravní náklady*, tj. náklady, spojené s přepravou jedné jednotky produktu. Cílem je uspokojit požadavky všech odběratelů při minimálních celkových přepravních nákladech, přičemž nesmí být překročena kapacita žádného z dodavatelů.

2.1 Matematický model

Příklad:

Firma vyrábějící bramborové lupínky zřizuje tři nové pobočky v Benešově, Jihlavě a Táboře. Hlavní surovinou jsou brambory, které se budou dovážet ze skladů v Humpolci a Pelhřimově. V tabulce 1 jsou uvedeny týdenní kapacity skladů a plánované týdenní požadavky výroben (v tunách). Přeprava brambor se bude uskutečňovat po železnici (jednou týdně). Tabulka 1 obsahuje jednotkové náklady na přepravu jedné tuny brambor od dodavatelů k odběratelům.

Tab. 1 Zadání dopravního problému

	Benešov	Jihlava	Tábor	Kapacita
Humpolec	330	250	350	70
Pelhřimov	300	240	250	80
Požadavek	45	60	35	

Cílem je naplánovat přepravu brambor tak, aby celkové přepravní náklady byly minimální.

Proměnné

Označíme-li dodavatele a odběratele uvedené v tabulce 1 jejich pořadovými čísly, zavedeme nezáporné proměnné se dvěma indexy:

x_{ij} = objem týdenní přepravy brambor (v tunách) od $i-$tého dodavatele k $j-$tému odběrateli $(i=1,2;\ j=1,2,3)$.

Účelová funkce

Účelová funkce představuje celkové týdenní přepravní náklady (v Kč):

$$z = 330x_{11} + 250x_{12} + 350x_{13} + 300x_{21} + 240x_{22} + 250x_{23} \to \min . \qquad (13)$$

Omezující podmínky

Při rozvozu brambor nesmí být překročeny kapacity dodavatelů, což vyjádříme následujícími nerovnicemi:

$$x_{11} + x_{12} + x_{13} \le 70\,, \tag{14}$$

$$x_{21} + x_{22} + x_{23} \le 80\,. \tag{15}$$

Splnění požadavků všech provozoven zajistí tyto rovnice:

$$x_{11} + x_{21} = 45\,, \tag{16}$$

$$x_{12} + x_{22} = 60\,, \tag{17}$$

$$x_{13} + x_{23} = 35\,. \tag{18}$$

Následují podmínky nezápornosti:

$$x_{ij} \ge 0, \qquad i = 1,2;\; j = 1,2,3. \tag{19}$$

Zavedení podmínek celočíselnosti opět závisí na reálné možnosti přepravovat brambory v menších jednotkách, v tomto případě kilogramech. Dopravní problém je však navíc speciálním typem úlohy (tzv. *unimodulární úloha*), v níž za předpokladu celočíselnosti kapacit a požadavků musí vždy vyjít celočíselné řešení. Bližší vysvětlení této vlastnosti je nad rámec základního kurzu operačního výzkumu a patří do oblasti složitějších kombinatorických metod.

2.2 Řešení úlohy v MS Excel

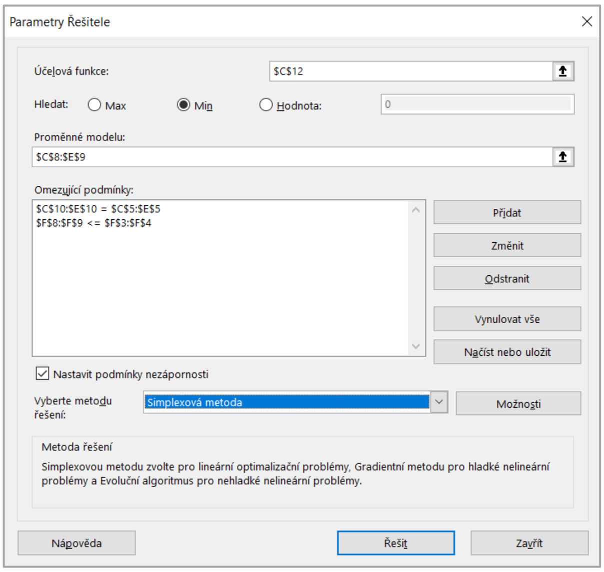
Protože má zadání DP jistá specifika (především dva indexy), provedeme výpočet optimálního řešení v *Řešiteli* MS Excel. Na Obr. 11 je uvedeno zadání úlohy v tabulce MS Excel. V horní části jsou zadány parametry úlohy, tedy kapacity dodavatelů, požadavky odběratelů a jednotkové přepravní náklady. V oblasti C8:E9 jsou buňky rezervované pro proměnné x_{ij}. Ve sloupci F pod označením Celkem jsou řádkové součty proměnných „=SUMA(C8:E8)“ a „=SUMA(C9:E9)“.

Podobně, v řádku 10 jsou sloupcové součty „=SUMA(C8:C9)“ atd. Ve všech případech se tedy jedná o levé strany omezujících podmínek (14) – (18). V buňce C12 je vzorec pro účelovou funkci „=SOUČIN.SKALÁRNÍ(C3:E4;C8:E9)“.

	A	B	C	D	E	F	G
1							
2			Benesov	Jihlava	Tabor	Kapacita	
3		Humpolec	330	250	350	70	
4		Pelhrimov	300	240	250	80	
5		Pozadavek	45	60	35		
6							
7		Preprava	Benesov	Jihlava	Tabor	Celkem	
8		Humpolec	0	0	0	0	
9		Pelhrimov	0	0	0	0	
10		Celkem	0	0	0		
11							
12		Naklady	0				
13							

Obr. 11 Zadání DP v tabulce MS Excel

Obr. 12 je vyplněné dialogové okno *Parametry Řešitele*. Je zřejmé, že v okně *Omezující podmínky* bylo nutné specifikovat dvě skupiny podmínek, zvlášť pro dodavatele a odběratele.



Obr. 12 Zadání matematického modelu DP v okně *Parametry Řešitele*

Na dalším obrázku (viz Obr. 13) je již výsledné optimální řešení. Jeho interpretace je jednoduchá. Buňky proměnných obsahují týdenní objemy přepravy (v tunách) mezi jednotlivými dodavateli a odběrateli. Velikost celkových týdenních nákladů je 37250 Kč, což je o 1000 Kč nižší hodnota než náklady získané metodou severozápadního rohu. Navíc lze z řešení snadno zjistit, že kapacita skladu v Humpolci nebude vyčerpána, zbyde v něm 10 tun (rozdíl mezi kapacitou 70 tun a množstvím, které se z něj odveze, tj. 60 tun). Tuto hodnotu lze najít i ve výsledkové zprávě ve sloupci *Odchylka*.

	A	B	C	D	E	F	G
1							
2			Benesov	Jihlava	Tabor	Kapacita	
3		Humpolec	330	250	350	70	
4		Pelhrimov	300	240	250	80	
5		Pozadavek	45	60	35		
6							
7		Preprava	Benesov	Jihlava	Tabor	Celkem	
8		Humpolec	0	60	0	60	
9		Pelhrimov	45	0	35	80	
10		Celkem	45	60	35		
11							
12		Naklady	37250				
13							

Obr. 13 Optimální řešení DP

2.3 Řešení úlohy v MPL for Windows

Obecná formulace matematického modelu dopravního problému

Při řešení složitějších úloh je výhodné zapisovat matematické modely obecně pomocí symbolů, tedy nikoli pomocí konkrétně zadaných parametrů. Jestliže je model určité úlohy formulovaný obecně, lze jej potom aplikovat na různé instance problému, a to dokonce i různého rozměru.

Předpokládejme, že v dopravním problému je dáno m dodavatelů, kteří nabízejí homogenní produkt n odběratelům. Kapacitu i – tého dodavatele (např. v tunách) označíme a_i $(i=1,2,...,m)$, požadavek j – tého odběratele (ve stejných jednotkách, tedy také v tunách) označíme b_j $(j=1,2,...,n)$. Jednotkové přepravní náklady (např. v Kč za tunu) mezi i – tým dodavatelem a j – tým odběratelem jsou označeny c_{ij}. Pro hledané množství produktu přepravované mezi i – tým dodavatelem a j – tým odběratelem, zavedeme nezápornou proměnnou x_{ij}.

Matematický model úlohy lze potom zapsat následujícím způsobem (využijeme zkrácený zápis pomocí sum):

$$z=\sum_{i=1}^{m}\sum_{j=1}^{n}c_{ij}x_{ij}\rightarrow \min, \tag{20}$$

$$\sum_{j=1}^{n}x_{ij}\le a_i, \quad i=1,2,...,m, \tag{21}$$

$$\sum_{i=1}^{m}x_{ij}=b_j, \quad j=1,2,...,n, \tag{22}$$

$$x_{ij}\ge 0, \quad i=1,2,...,m; j=1,2,...,n. \tag{23}$$

Řešení dopravního problému pomocí MPL

Význam obecné formulace matematického modelu úlohy si ukážeme na řešení dopravního problému pomocí MPL. Profesionální přístup k řešení úloh LP s využitím specializovaných modelovacích systémů většinou předpokládá, že v těchto systémech je zapsán obecný matematický model a konkrétní data jsou do něj importována např. z MS Excel či textového souboru. Ideálním završením celého procesu řešení je pak export výsledků do prostředí, které je známé vlastnímu zadavateli úlohy, tedy např. opět MS Excel. To bude i náš případ.

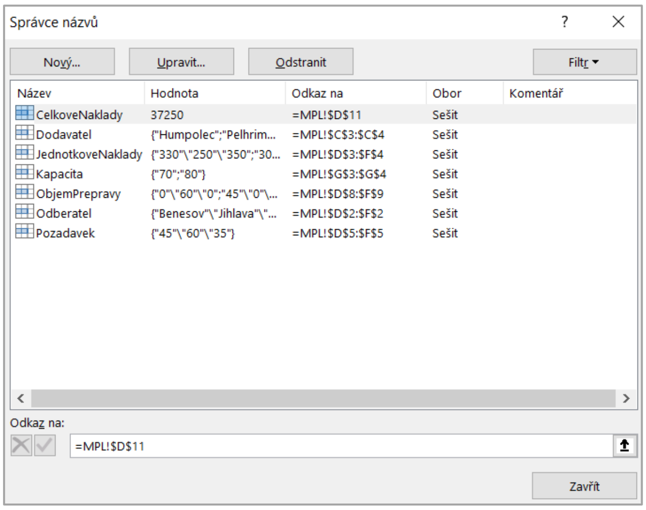
V první řadě je nutné v tabulce MS Excel připravit soubor se vstupními daty. Lze použít stejný sešit, v němž byla úloha vyřešena pomocí *Řešitele* MS Excel. Abychom v souboru zachovali i toto řešení, zkopírujeme všechna data i s řešením na nový list, který nazveme „MPL“. Celý soubor uložíme pod názvem „DopravniProblem.xlsx“. Při zadávání všech názvů jak v MS Excel, tak v MPL, je nutné dodržet zásadu nepoužívat diakritiku, mezery, ani žádné speciální znaky kromě podtržítka.

Na Obr. 14 je uvedena tabulka vstupních dat spolu s oblastí pro výstup (pro jasnější identifikaci jsou obě oblasti graficky odděleny. Lze si povšimnout, že v porovnání s listem na Obr. 11 chybí řádkové a sloupcové součty. Navíc v buňce celkových nákladů není vložena funkce skalárního součinu jednotkových přepravních nákladů a objemů přepravy, buňka je tedy prázdná.

	A	B	C	D	E	F	G	H
1								
2	Vstup			Benesov	Jihlava	Tabor	Kapacita	
3			Humpolec	330	250	350	70	
4			Pelhrimov	300	240	250	80	
5			Pozadavek	45	60	35		
6								
7	Výstup		Preprava	Benesov	Jihlava	Tabor		
8			Humpolec	0	0	0		
9			Pelhrimov	0	0	0		
10								
11			Naklady					
12								

Obr. 14 Příprava sešitu MS Excel pro MPL

Pro komunikaci MPL s tímto vstupně-výstupním souborem je nutné pojmenovat jednotlivé oblasti[7] tak, jak je to uvedeno ve *Správci názvů* na kartě *Vzorce*. Finální seznam všech názvů je uveden na Obr. 15.



Obr. 15 Seznam pojmenovaných oblastí

Jak bylo výše uvedeno, zadání matematického modelu v MPL (viz Obr. 16) vychází z jeho obecné formulace (20) – (23) a podstatně se liší od jednoduchého způsobu zadání konkrétního modelu, který byl použit výše pro řešení jednoduchých úloh LP. Jednotlivé dříve definované části modelu jsou nyní definovány na první pohled složitějším způsobem, některé sekce jsou zcela nové. K nim patří sekce *OPTIONS*, která obsahuje hned dvě položky, nutné pro komunikaci s výše definovaným sešitem. Název sešitu je uveden v první položce. Ve druhé položce je uveden název příslušného listu. Tento název není nutné uvádět, pokud se v sešitu jedná o první list v pořadí.

```
TITLE DopravniProblem;

OPTIONS
ExcelWorkBook="DopravniProblem.xlsx";
ExcelSheetName="MPL";

INDEX
i:=EXCELRANGE("Dodavatel");
j:=EXCELRANGE("Odberatel");

DATA
a[i]:=EXCELRANGE("Kapacita");
b[j]:=EXCELRANGE("Pozadavek");
c[i,j]:=EXCELRANGE("JednotkoveNaklady");

VARIABLES
x[i,j] EXPORT TO EXCELRANGE("ObjemPrepravy");

MODEL

MIN
z EXPORT TO EXCELRANGE("CelkoveNaklady") =sum(c*x);

SUBJECT TO
dodavatele[i]:sum(j:x[i,j])<=a[i];
odberatele[j]:sum(i:x[i,j])=b[j];

END
```

Obr. 16 Zápis obecného modelu DP v MPL

Další novou sekcí je sekce INDEX. V ní jsou definovány všechny použité indexy. Index i označuje dodavatele, index j odběratele. Správně by měl být na začátku určen počet dodavatelů a odběratelů, tedy obecně použito symbolů m a n. Tady se ukazuje výhoda zadání vstupních dat v MS Excel (pro identifikaci oblasti používáme klíčové slovo EXCELRANGE).

[7] Pro pojmenování oblastí je možné využít *Pole názvů*. V případě, že se při zadávání názvu dopustíme chyby, je nutné ji odstranit ve *Správci názvů*.

Pokud v sešitu definujeme oblast dodavatelů a odběratelů, pak jejich počet je dán automaticky velikostí příslušných oblastí a není nutné se počtem dodavatelů a odběratelů v konkrétní instanci zabývat. V tom spočívá kouzlo profesionálního využití specializovaného softwaru.

Část DATA je určena k načtení vstupních dat do jednotlivých obecně označených parametrů. Zde je důležité v hranatých závorkách použít správné indexy. Pro odkaz na správný název oblasti v tabulce MS Excel je opět použité klíčové slovo EXCELRANGE.

Také u definice proměnných v části VARIABLES je nutné využít správné indexy. Jestliže chceme mít v sešitu uvedeny výsledky optimalizace, je v něm nutné definovat název příslušné oblasti, v tomto případě oblasti s objemy přepravy. Pro samotný výstup se pak použije příkaz EXPORT TO, jak je zřejmé z Obr. 16.

V matematickém modelu hraje velice důležitou roli funkce „sum“, pomocí níž lze zapsat účelovou funkci a levé strany omezujících podmínek. Její syntaxe je následující: „sum(*seznam indexů*: *sčítanec*)“. Seznam indexů odpovídá jednotlivým sumacím, resp. sumačním indexům, indexy jsou odděleny čárkou. Sčítanec je člen sumace, který je samozřejmě vyjádřen pomocí sumačních indexů. Jistou specialitou MPL je zjednodušený zápis funkce, kdy není nutné zadávat seznam indexů a zapisovat indexy u sčítanců, pokud je zápisem jednoznačně určeno, co a jak se sčítá. Zápis účelové funkce na Obr. 16 je zkrácenou verzí úplného zápisu „sum(i,j:c[i,j]*x[i,j])“. V modelu je na začátku řádku uveden název účelové funkce, v našem případě jsme ponechali obvyklé označení „z“. Následuje příkaz pro export optimální hodnoty do MS Excel (samozřejmě pokud existuje optimální řešení) a samotný zápis účelové funkce uvozený rovnítkem.

Zřejmě nejsložitější částí je zápis omezujících podmínek. Nejprve budou definovány podmínky pro dodavatele, tedy podmínky (21). V úvodu je nutné se věnovat indexu, který je zaveden právě pro dodavatele, což je index i. Tím je totiž určeno, kolik nerovností tohoto typu bude zápis představovat. V modelu je uveden počet m, z předchozího textu je zřejmé, že v MPL se odkazujeme na část INDEX, kde je index i definován. Na Obr. 16 zápis „dodavatele[i]“ odpovídá tedy v podmínkách (21) zápisu „$i = 1, 2, \ldots, m$“, přičemž samotný název „dodavatele“ je libovolným názvem omezujících podmínek a měl by vždy logicky vyjadřovat, čeho se dané podmínky týkají. Za dvojtečkou pak následuje levá strana omezující podmínky. V tomto případě se jedná o součet „sum(j:x[i,j])“, kde sumačním indexem je j a sčítancem jsou jednotlivé proměnné. Tento zápis v podstatě „doslova“ kopíruje zápis levé strany podmínek (21). Totéž se týká typu omezující podmínky „<=“ a pravé strany „a[i]“. Podobným způsobem lze rozšifrovat zápis omezujících podmínek pro odběratele.

Vzhledem k tomu, že jsme v modelu použili příkazy pro export výsledků do tabulky MS Excel, po dokončení optimalizace se přímo v sešitu objeví optimální řešení spolu s optimální hodnotou účelové funkce (viz Obr. 17). Na Obr. 18 je totéž řešení získané v podobě výstupní sestavy přímo v prostředí MPL.

6						
7	Výstup		Preprava	Benesov	Jihlava	Tabor
8			Humpolec	0	60	0
9			Pelhrimov	45	0	35
10						
11			Naklady	37250		
12						

Obr. 17 Export optimálního řešení do sešitu MS Excel

```
SOLUTION RESULT

  Optimal solution found

    MIN z                =      37250.0000

DECISION VARIABLES

VARIABLE x[i,j] :

   i          j                Activity     Reduced Cost
  -------------------------------------------------------
   Humpolec   Benesov            0.0000          20.0000
   Humpolec   Jihlava           60.0000           0.0000
   Humpolec   Tabor              0.0000          90.0000
   Pelhrimov  Benesov           45.0000           0.0000
   Pelhrimov  Jihlava            0.0000           0.0000
   Pelhrimov  Tabor             35.0000           0.0000
  -------------------------------------------------------

CONSTRAINTS

CONSTRAINT dodavatele[i] :

   i                    Slack      Shadow Price
  ----------------------------------------------
   Humpolec           10.0000            0.0000
   Pelhrimov           0.0000          -10.0000
  ----------------------------------------------

CONSTRAINT odberatele[j] :

   j                  Slack      Shadow Price
  --------------------------------------------
   Benesov           0.0000          310.0000
   Jihlava           0.0000          250.0000
   Tabor             0.0000          260.0000
```

Obr. 18 Optimální řešení v MPL

2.4 Kontejnerový dopravní problém

Zatímco v klasickém dopravním problému je cílem určit objemy přepravy mezi jednotlivými dodavateli a odběrateli, v kontejnerovém dopravním problému (KDP) je pro přepravu důležitý i její způsob. K přepravě se používají kontejnery o stejné kapacitě, přičemž přepravní náklady nejsou vztaženy na přepravovanou jednotku, ale na využití jednoho kontejneru, přepravovaného mezi dodavatelem a odběratelem.

Příklad:

Předpokládejme, že v předešlém příkladu si bude přepravní společnost účtovat ceny za přepravu (pronájem) jednoho vagónu mezi jednotlivými dodavateli a odběrateli podle tabulky 2. Kapacity dodavatelů a požadavky odběratelů jsou shodné s hodnotami, zadanými v předešlém příkladu. K přepravě lze použít vagóny o kapacitě 18 tun. Cílem je jednak určit, kolik tun brambor se bude přepravovat mezi jednotlivými místy, ale také stanovit, kolik vagónů bude na tuto přepravu použito, aby i v tomto případě byly celkové přepravní náklady minimální.

Tab. 2 Zadání kontejnerového dopravního problému

	Benešov	Jihlava	Tábor	Kapacita
Humpolec	4200	4800	5300	70
Pelhřimov	5100	3400	3700	80
Požadavek	45	60	35	

Protože KDP a DP mají mnoho společného, soustředíme se především na rozdíly v zadání obou úloh a formulaci jejich obecných matematických modelů. Vzhledem k jednoduchosti úlohy se budeme zabývat přímo obecnou formulací matematického modelu.

V KDP zavedeme stejné označení jednotlivých parametrů jako v DP, pokud jde o počet dodavatelů a odběratelů, kapacity dodavatelů a požadavky odběratelů. Parametr c_{ij} bude tentokrát použitý pro náklady (v Kč) na přepravu jednoho kontejneru mezi i – tým dodavatelem

$(i=1,2,...,m)$ a j – tým odběratelem $(j=1,2,...,n)$. Symbolem K označíme kapacitu kontejneru ve stejných jednotkách jako jsou kapacity a požadavky, tedy v našem případě v tunách.

Proměnné

Kromě proměnných x_{ij}, vyjadřujících objem přepravy (v tunách) od i – tého dodavatele k j – tému odběrateli, je nutné zavést další proměnné:

y_{ij} = počet kontejnerů, které budou použity k přepravě od i – tého dodavatele k j – tému odběrateli $(i=1,2,...,m;\ j=1,2...,n)$.

Účelová funkce

Protože se přepravní náklady c_{ij} vztahují na přepravu jednoho kontejneru mezi i – tým dodavatelem a j – tým odběratelem, účelová funkce je formulována následovně:

$$z=\sum_{i=1}^{m}\sum_{j=1}^{n}c_{ij}y_{ij}\rightarrow \min . \qquad (24)$$

Omezující podmínky

Omezující podmínky (21) a (22), stejně jako podmínky nezápornosti (23), jsou nezměněné. Navíc ovšem přidáme bilanční podmínky mezi objemem přepravy a počtem použitých kontejnerů:

$$x_{ij}\le Ky_{ij}, \quad i=1,2,...,m;\ j=1,2,...,n. \qquad (25)$$

Význam podmínek je následující. Bude-li proměnná x_{ij} kladná, pak počet použitých kontejnerů y_{ij} musí být takový, aby bylo možné daný objem přepravit. Jestliže tedy v našem příkladu je objem přepravy 40 tun, pak při kapacitě vagónu 18 tun bude nastaven počet kontejnerů na hodnotu 3. Samozřejmě by nerovnice (25) byla splněna i pro větší počet kontejnerů, cílem je ovšem minimalizovat náklady na přepravu. Proto se bude používat nejnižší možný počet kontejnerů. Podmínky (25) mají ještě další funkci. Pokud se mezi danou dvojicí dodavatele a odběratele nebude používat žádný kontejner, tedy $y_{ij}=0$, pak hodnota proměnné x_{ij} musí být také nulová, což znamená, že mezi danou proměnnou se nic přepravovat nesmí.

Navíc je nutné přidat podmínky nezápornosti a celočíselnosti pro novou proměnnou:

$$y_{ij}\ge 0,\ \text{celé}, \quad i=1,2,...,m;\ j=1,2,...,n. \qquad (26)$$

Řešení KDP

Pro řešení KDP v *Řešiteli* Excel je nutné zopakovat (z podkapitoly 2.3) důležitý postup, jak zadat v okně *Parametry Řešitele* dvě nesouvislé oblasti buněk proměnných. Jedna oblast je totiž vyhrazena pro objemy přepravy, druhá oblast pro počty kontejnerů. Při výběru oblastí myší je nutné použít klávesu CTRL. Jednotlivé adresy oblastí jsou pak v rámečku proměnných odděleny středníkem.

V tabulkách 3 a 4 je uvedeno optimální řešení příkladu. Při interpretaci výsledků zjistíme, že v Humpolci tentokrát zůstane nevyužito 7 tun brambor, v Pelhřimově 3 tuny. Vezmeme-li v úvahu zadanou kapacitu vagónu 18 tun, pak třetí vagón, směřující z Humpolce do Benešova, bude vytížen přesně z jedné poloviny. Přepravu lze samozřejmě naplánovat i tak, že všechny tři vagóny budou rovnoměrně vytíženy objemem 15 tun. Jediný vagón, směřující z Humpolce do Jihlavy, bude plně vytížen. Ve třetím vagónu, jedoucím z Pelhřimova do Jihlavy, bude naloženo pouze 6 tun, tj. bude obsazena jedna třetina jeho celkové kapacity, případně lze opět rozložit náklad rovnoměrně po 14 tunách. První vagón z Pelhřimova do Tábora bude zcela naplněn, ve druhém bude naloženo 17 tun brambor. Optimální přepravní náklady činí 35000 Kč.

Tab. 3 Objemy přepravy

	Benešov	Jihlava	Tábor
Humpolec	45	18	-
Pelhřimov	–	42	35

Tab. 4 Počty vagónů

	Benešov	Jihlava	Tábor
Humpolec	3	1	-
Pelhřimov	–	3	2

Při řešení KDP v MPL je pro definici proměnných y_{ij} nutné přidat další část INTEGER VARIABLES. Podobně, pro definici binárních proměnných, které se vyskytují např. v přiřazovacích problémech či v úloze obchodního cestujícího, je nutné přidat část BINARY VARIABLES.

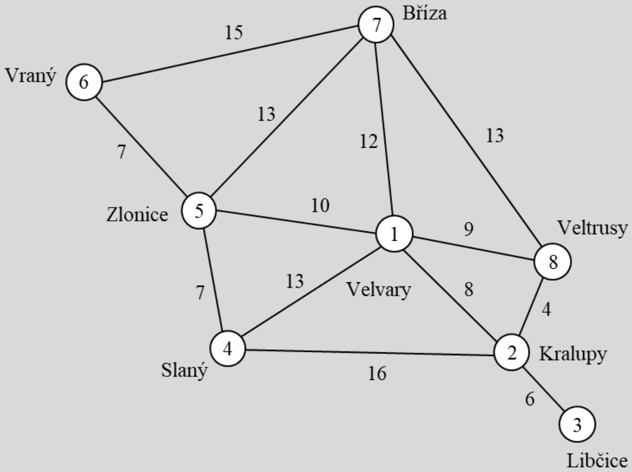
3 Úloha obchodního cestujícího

Tato úloha se někdy nazývá *okružní dopravní problém*. V angličtině se používá název *Traveling Salesman Problem*, v dalším textu budeme pro tuto úlohu používat vžitou zkratku tohoto názvu TSP. Zadání úlohy je poměrně jednoduché. Obchodní cestující, resp. vozidlo má navštívit na trase několik míst, a to každé místo právě jednou, a vrátit se zpět do výchozího místa. Výsledná trasa tudíž tvoří okruh (cyklus) a cílem je nalézt nejkratší takový okruh.

P

Příklad:

Obchodní zástupce pivovaru ve Velvarech musí postupně navštívit 7 restaurací v 7 obcích. Na Obr. 19 jsou znázorněny silnice, po nichž je možné mezi obcemi jet. V tabulce 5 jsou uvedeny vzdálenosti (v km) odpovídající přímým spojením obcí (silnicemi). Pomlčka odpovídá situaci, kdy obce nejsou přímo spojeny silnicí. Cílem je navštívit všechny restaurace a ujet přitom minimální vzdálenost.



Obr. 19 Distribuční mapa se zákazníky

Tab. 5 Přímá spojení obcí

	Velv	Kra	Lib	Sla	Zlo	Vra	Bri	Velt
Velvary	0	8	-	13	10	-	12	9
Kralupy	8	0	6	16	-	-	-	4
Libcice	-	6	0	-	-	-	-	-
Slany	13	16	-	0	7	-	-	-
Zlonice	10	-	-	7	0	7	13	-
Vrany	-	-	-	-	7	0	15	-
Briza	12	-	-	-	13	15	0	13
Veltrusy	9	4	-	-	-	-	13	0

Pro vyřešení úlohy je nutné nejprve sestavit matici vzdáleností mezi dvojicí míst[8] (viz tabulka 6).

[8] K tomu lze použít celou řadu nástrojů, s uvedením přesné adresy místa např. server https://mapy.cz nebo https://www.google.com/maps. Hledání nejkratších cest mezi dvojicí míst bude věnována také část následující kapitoly.

Tab. 6 Matice vzdáleností mezi místy

	Velv	Kra	Lib	Sla	Zlo	Vra	Bri	Velt
Velvary	0	8	14	13	10	17	12	9
Kralupy	8	0	6	16	18	25	17	4
Libcice	14	6	0	22	24	31	23	10
Slany	13	16	22	0	7	14	20	20
Zlonice	10	18	24	7	0	7	13	19
Vrany	17	25	31	14	7	0	15	26
Briza	12	17	23	20	13	15	0	13
Veltrusy	9	4	10	20	19	26	13	0

Pro nalezení optimálního řešení lze formulovat matematický model. Obecně lze úlohu zadat tak, že existuje n míst (výchozí místo a $n-1$ zákazníků), jejichž vzdálenosti jsou označeny jako c_{ij}. Často je úloha symetrická, tj. platí $c_{ij} = c_{ji}$ pro všechna $i = 1, 2, ..., n$ a $j = 1, 2, ..., n$. Výchozí obec je označena číslem 1.

Proměnné

V matematickém modelu zavedeme dva typy proměnných:

$$x_{ij} = \begin{cases} 1 & \text{jestliže vozidlo jede přímo z místa } i \\ & \text{do místa } j, \qquad\qquad i = 1, 2, ..., n, \\ & \qquad\qquad\qquad\qquad j = 1, 2, ..., n, \\ 0 & \text{jinak,} \end{cases} \tag{27}$$

$u_i =$ umělá proměnná zavedená v podmínkách zabraňujících vytváření parciálních cyklů $(i = 1, 2, ..., n)$, vysvětlení viz dále.

Účelová funkce

Účelová funkce představuje celkovou délku trasy:

$$z = \sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \rightarrow \min . \tag{28}$$

Omezující podmínky

Důležitou podmínkou v TSP je nutnost navštívit každé místo právě jednou. V modelu pro tuto podmínku použijeme dvě soustavy rovnic. Soustava (29) zajišťuje, že z každého místa vozidlo vyjede právě jednou, podmínky (30) určují, že do každého místa vjede vozidlo právě jednou:

$$\sum_{j=1}^{n} x_{ij} = 1, \quad i = 1, 2, ..., n , \tag{29}$$

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, 2, ..., n . \tag{30}$$

Výše uvedené soustavy rovnic připomínají omezující podmínky přiřazovacího problému. To je ostatně patrné i z řešení v tabulce 7, kde v každém řádku a v každém sloupci je právě jedno označené políčko stejně jako právě v přiřazovacím problému. Ne každé takové řešení je ovšem přípustným řešením TSP. V tabulce 7 je uvedeno takové řešení, které splňuje podmínky (29) a (30), nikoli však původní podmínky TSP.

Tab. 7 Nepřípustné řešení TSP

	Velv	Kra	Lib	Sla	Zlo	Vra	Bri	Velt
Velvary	0	8	14	13	10	17	12	9
Kralupy	8	0	6	16	18	25	17	4
Libcice	14	6	0	22	24	31	23	10
Slany	13	16	22	0	7	14	20	20
Zlonice	10	18	24	7	0	7	13	19
Vrany	17	25	31	14	7	0	15	26
Briza	12	17	23	20	13	15	0	13
Veltrusy	9	4	10	20	19	26	13	0

Z tabulky je množné vyčíst, že se v řešení nejedná o jediný okruh, ale o dva samostatné okruhy. V první trase vozidlo navštíví místa v pořadí Velvary, Kralupy, Veltrusy, Libčice a Velvary, ve druhé trase např. v pořadí Slaný, Zlonice, Vraný, Bříza a Slaný. U druhé trasy není jednoznačně určeno, které místo je výchozí. Řešení se rozpadlo na dvě trasy, tzv. *parciální cykly*. Abychom zabránili vzniku takových okruhů, je nutné do modelu zahrnou následující podmínky:

$$u_i + 1 - (n-1)(1-x_{ij}) \le u_j, \quad i = 1,2,\ldots,n;\ j = 2,3,\ldots,n\,. \qquad (31)$$

V soustavě nerovnic jsou zavedeny proměnné u_i, jejichž hodnoty se v průběhu okruhu postupně zvyšují o hodnotu 1. Z této podmínky je vynechán jediný přejezd, a sice návrat z posledního místa na trase do výchozího místa (dolní hranice indexu j je 2, nikoli 1). Proč se hodnoty u_i zvyšují právě o hodnotu 1 lze vysvětlit následujícím způsobem. Uskuteční-li se přejezd z místa i do místa j, tj., podmínka (31) bude mít tvar $u_i + 1 \le u_j$, proměnná se tedy musí zvýšit alespoň o hodnotu 1. Pro $x_{ij} = 0$ bude mít podmínka (31) po úpravě tvar $u_i - u_j \le n-2$, což znamená, že rozdíl hodnot jakýchkoliv dvou uvedených proměnných nepřesáhne hodnotu $n-2$. Nesmíme zapomenout, že v soustavě nerovnic nejsou podmínky pro $j = 1$. Pokud je tedy $n = 8$ jako v našem příkladu, pak rozdíl hodnot proměnných je maximálně 6. Z toho tedy jasně vyplývá, že nejvyšší hodnota je 7 a nejnižší 1 (pokud samozřejmě neuvažujeme místo 1, pro které je $u_1 = 0$).

Lze se snadno přesvědčit, že první trasa v nepřípustném řešení v tabulce 7 soustavu podmínek (31) splňuje, druhá trasa však podmínky porušuje. Pokud by např. proměnná u_4 byla rovna 0, pak $u_5 = 1$, $u_6 = 2$ a $u_7 = 3$. Protože se uskutečňuje i návrat do místa Slaný, tedy přejezd z místa 7 do místa 4, měla by proměnná u_4 nabývat hodnoty 8, což ovšem nemůže, protože již nabývá hodnoty 0.

V matematickém modelu nesmíme zapomenout následující podmínky:

$$x_{ij} \in \{0,1\}, \quad i = 1,2,\ldots,n;\ j = 1,2,\ldots,n, \qquad (32)$$

$$u_i \ge 0, \quad i = 1,2,\ldots,n\,. \qquad (33)$$

Pozn. Pokud bychom k řešení TSP využili modelovací systém MPL, pak je zapotřebí upozornit na tři skutečnosti:

- Jelikož pro místa bude použitý index i, ale zároveň index j, je nutné, podobně jako v úloze o pokrytí, použít následující definici indexů:
  INDEX
  i:=EXCELRANGE("Misto");
  j:=i;
- Protože se snažíme i v MPL zapisovat modely obecně, je nutné zjistit počet míst přímo z dat, která budou zadána ve vstupním sešitu MS Excel. K tomuto účelu je možné použít v sekci DATA následující funkci:
  n:=count(i);
- Protože MPL překvapivě neumožňuje násobit dvě závorky, je nutné pro zápis cyklických podmínek použít tento zápis:
  u[i]+1-n*(1-x[i,j])<=u[j];
  přičemž místo výše uvedeného zápisu n:=count(i); zapíšeme v sekci DATA
  n:=count(i) – 1;

Optimální řešení

V tabulce 8 je vyznačeno optimální řešení příkladu, a to včetně hodnot u_i. Minimální délka trasy činí 79 km.

Tab. 8 Optimální řešení TSP

	Velv	Kra	Lib	Sla	Zlo	Vra	Bri	Velt	u_i
Velvary	0	8	14	13	10	17	12	9	0
Kralupy	8	0	6	16	18	25	17	4	6
Libcice	14	6	0	22	24	31	23	10	7
Slany	13	16	22	0	7	14	20	20	1
Zlonice	10	18	24	7	0	7	13	19	2
Vrany	17	25	31	14	7	0	15	26	3
Briza	12	17	23	20	13	15	0	13	4
Veltrusy	9	4	10	20	19	26	13	0	5