



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MATEMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF MATHEMATICS

OPTIMALIZAČNÍ MODELY V LOGISTICE

OPTIMIZATION IN LOGISTICS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. ALENA HUCLOVÁ

VEDOUCÍ PRÁCE

SUPERVISOR

RNDr. PAVEL POPELA, Ph.D.

BRNO 2010

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Ústav matematiky

Akademický rok: 2009/2010

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Alena Huclová

který/která studuje v **magisterském navazujícím studijním programu**

obor: **Matematické inženýrství (3901T021)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Optimalizační modely v logistice

v anglickém jazyce:

Optimization in Logistics

Stručná charakteristika problematiky úkolu:

Student si prohloubí znalosti problematiky matematického programování ve vztahu k vybraným problémům logistiky se zaměřením na matematické modely v podmínkách neurčitosti.

Důraz bude kladen na modely stochastického programování.

Student se bude zabývat scénářovými přístupy.

Cíle diplomové práce:

Vybraný problém optimalizace v logistice bude modelován a řešen s využitím optimalizačního software GAMS.

Seznam odborné literatury:

Popela, P. Stochastic Programming, UoM, 2004

Kall, P. - Wallace, S.W. Stochastic Programming, Wiley, 1994

Laporte, F. Optimization in Logistics, Wiley 2003

Vedoucí diplomové práce: RNDr. Pavel Popela, Ph.D.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2009/2010.

V Brně, dne 20.11.2009

L.S.

prof. RNDr. Josef Šlapal, CSc.
Ředitel ústavu

prof. RNDr. Miroslav Doupovec, CSc.
Děkan fakulty

Abstrakt

Práce je zaměřena na optimalizaci modelů dopravního a přepravního problému s náhodností poptávky, přidáním hran a dynamickým oceňováním. V teoretické části práce jsou uvedeny matematické modely dopravy a popsán software GAMS, který je při řešení použit. Praktická část průběžně doplňuje teoretickou část a aplikuje popsané modely na reálná data.

Klíčová slova

Optimalizace, teorie grafů, lineární programování, dopravní problém, přepravní problém, stochastické programování, náhodnost poptávky, celočíselné proměnné, dynamické oceňování, GAMS

Abstract

The thesis is focused on the optimization of models of transportation and transshipment problem with random demand, additional edges, and dynamic pricing. The theoretical part of the thesis introduces mathematical models of transportation. The software GAMS, which is used for the solution, is all so described. The practical part is a split among chapters and implements the described models by using real data.

Key words

Optimization, graph theory, linear programming, transportation problem, transshipment problem, stochastic programming, random demand, integer variables, dynamic pricing, GAMS

Bibliografická citace

HUCLOVÁ, A. *Optimalizační modely v logistice*. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2010. 61 s. Vedoucí diplomové práce RNDr. Pavel Popela, Ph.D.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma „Optimalizační modely v logistice“ jsem vypracovala samostatně pod vedením vedoucího diplomové práce RNDr. Pavla Popely, Ph.D. a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

V Brně dne 28.5.2010

Bc. Alena Huclová

Poděkování

Děkuji vedoucímu diplomové práce RNDr. Pavlu Popelovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

Obsah

1	Úvod.....	6
1.1	Základní pojmy z teorie grafů.....	6
1.2	Lineární programování.....	8
2	Dopravní úloha.....	9
2.1	Formulace úlohy.....	9
2.2	Matematický model.....	10
2.3	GAMS.....	11
2.4	Příklad 2.1.....	11
2.5	Příklad 2.2.....	15
3	Přepravní úloha.....	18
3.1	Formulace speciální úlohy.....	18
3.2	Matematický model.....	18
3.3	Příklad 3.1 – Speciální přepravní problém.....	19
3.4	Obecný přepravní problém.....	20
3.5	Implementace příkladu 3.1 v GAMSu.....	21
4	Aplikační příklad.....	23
4.1	Aplikace dopravního modelu.....	24
4.2	Aplikace přepravního modelu.....	27
5	Problém náhodnosti poptávky.....	32
5.1	Stochastické programování.....	32
5.2	„Wait – and – see“ přístup (WS).....	32
5.3	„Here – and – now“ přístup (HN).....	32
5.4	Scénářové úlohy.....	33
5.5	Formulace úlohy.....	33
5.6	Matematický model.....	33
5.7	Aplikační příklad.....	35
6	Problém přidání hran.....	38
6.1	Formulace úlohy.....	38
6.2	Matematický model.....	38
6.3	Aplikační příklad.....	39
7	Variabilita poptávky i počtu hran.....	44
7.1	Formulace úlohy.....	44
7.2	Matematický model.....	44
7.3	Zobecněná formulace úlohy.....	45

7.4	Zobecněný matematický model.....	46
7.5	Aplikační příklad.....	46
8	Dynamické oceňování.....	50
8.1	Lineární model	50
8.2	Formulace úlohy.....	51
8.3	Matematický model.....	51
8.4	Aplikační příklad.....	52
9	Závěr	58
	Seznam použitých zdrojů.....	59
	Seznam použitých symbolů	60

1 Úvod

Logistika se zabývá organizací, přesunem a skladováním materiálu a lidí. Jejím cílem je zajistit zákazníkům správný produkt, na správném místě a ve správném čase, optimalizovat celkovou cenu (maximalizovat výnosy a minimalizovat náklady) a splnit daná omezení (např. omezení rozpočtu).

Potřeba organizovat zásobování a jejich toky se nejprve objevila v armádě. První náznaky se objevily již ve starém Řecku, Římě a Byzanci. Existovali důstojníci, kteří byli zodpovědní právě za ubytování a zásobování. V moderním válečnictví se důležitost logistiky ukázala naplno. Frontu bylo potřeba zásobovat střelivem, potravinami, lidmi i zbraněmi.

Do obchodní sféry pak přešla logistika v padesátých letech v USA, jako snaha dále snížit náklady. Systém byl vytvořen analogicky podle vojenského modelu, který se ukázal úspěšný a rozhodl mnoho bitev, viz [16] a [11].

Ve své práci se budu zabývat optimalizací dopravního a přepravního problému s modifikacemi. Těmi bude náhodnost poptávky, optimalizace dopravy a proměnlivost ceny. Budu přitom využívat modelů matematického programování a poznatků včetně grafického zobrazení z teorie grafů pro jejich názornost a přehlednost.

1.1 Základní pojmy z teorie grafů

Teorie grafů je poměrně mladá matematická disciplína, která je součástí diskretní matematiky. Tradičně se za zakladatele teorie grafů považuje švýcarský matematik Leonhard Euler (1707-1783), který roku 1736 ve své práci *Solutio problematis ad geometriam situs pertinentis* řešil úlohu, jak projít přes sedm mostů města Královce (Königsbergu) (každý z nich právě jednou) a vrátit se do výchozího místa.

Grafem v teorii grafů se rozumí objekty popsané množinou vrcholů a množinou hran, uvedme nyní základní pojmy a značení:

Jednoduchý graf

Graf G je uspořádaná dvojice $G = (V, E)$, kde V je neprázdná množina vrcholů a E je množina hran, tj. dvouprvkových podmnožin množiny V , viz [4], [9].

Grafy nejčastěji znázorňujeme diagramy, ve kterých vrcholy jsou reprezentovány kroužky a hrany jsou vyjádřeny spojnicemi dvojice vrcholů.

Vrchol v diagramu spojený hranou nazýváme *koncovým vrcholem* dané hrany a říkáme, že je *incidentní* s danou hranou. Koncové vrcholy téže hrany nazýváme *sousedními vrcholy*.

Úplný graf (také *kompletní*) má všechny vrcholy navzájem pospojované hranami.

Jsou-li dva vrcholy spojeny více než jednou hranou, mluvíme o *násobných hranách*. V případě, že jsou oba vrcholy spojené hranou totožné, hranu nazýváme *smyčkou*.

Graf, ve kterém nepřipouštíme násobné hrany ani smyčky, se nazývá *jednoduchý (prostý)* a s těmito grafy budeme nadále pracovat. V opačném případě se jedná o *multigraf*.

Graf, jehož hrany a (nebo) vrcholy jsou opatřeny číselnými (nebo jinými hodnotami), nazýváme *ohodnoceným grafem*. Ve své práci bud používat ohodnocené grafy.

Cesty a cykly

Sled je posloupnost $v_0, e_1, v_1, e_2, \dots, e_n, v_n$, kde $v_i \in V, e_i \in E, n \in \mathbf{N}, n \geq 1, i = 1, \dots, n$, kde e_i jsou hrany s koncovými vrcholy v_{i-1} a v_i . Ve sledu se mohou hrany i vrcholy opakovat.

Tah je sled, ve kterém se žádná hrana neopakuje. *Cesta* je tah, ve kterém se neopakuje žádný vrchol. Přidáme-li k cestě hranu e_{n+1} , jejíž koncové vrcholy jsou v_n a v_0 , dostaneme *cyklus (kružnici)*. Graf nezmene *acyklický*, jestliže žádný jeho podgraf není cyklem.

Dosažitelnost a souvislost grafu

Vrchol v je *dosažitelný* z vrcholu u , jestliže v grafu G existuje cesta z vrcholu u do vrcholu v , nebo pokud $u = v$.

Graf, ve kterém jsou z libovolného vrcholu dosažitelné všechny ostatní vrcholy, se nazývá *souvislý graf*.

Komponenta grafu je maximální souvislý podgraf daného grafu G .

Stromy a kostry

Pojmem *strom* označujeme acyklický a souvislý graf. *Les* je pak acyklický graf, jehož každá komponenta je strom.

Kostrou grafu G rozumíme podgraf v G , který je lesem, obsahuje všechny vrcholy grafu G a na každé souvislé komponentě G indukuje strom.

Orientovaný graf

Orientovaným grafem G nazýváme uspořádanou dvojici (V, E) , kde V je množina vrcholů a E je množina uspořádaných dvojic vrcholů, zvaných orientované hrany.

O *orientované hraně* $e = uv, e \in E; u, v \in V$ říkáme, že má počáteční vrchol u a koncový vrchol v . Dvě hrany uv a vu se stejnými koncovými vrcholy, avšak opačně orientované, nazýváme *opačné*.

Negrafičké zobrazení grafu

Incidenční matice $A^I(G)$ orientovaného grafu G s n vrcholy u_1, u_2, \dots, u_n a m hranami h_1, h_2, \dots, h_m je matice typu $n \times m$ s prvky a_{ij} , kde $i = 1, 2, \dots, n; j = 1, 2, \dots, m$

$$a_{ij} = \begin{cases} 1, & \text{jestliže } u_i \text{ je koncový vrchol hrany } h_j \\ -1, & \text{jestliže } u_i \text{ je počáteční vrchol hrany } h_j \\ 0 & \text{jinak} \end{cases}$$

Matice sousednosti $A^S(G)$ orientovaného grafu G s n vrcholy u_1, u_2, \dots, u_n je čtvercová matice $n \times n$ s prvky a_{ij} , kde $i, j = 1, 2, \dots, n$

$$a_{ij} = \begin{cases} 1, & \text{jestliže vrcholy od } u_i \text{ k } u_j \text{ jsou sousední} \\ 0, & \text{jinak} \end{cases}$$

1.2 Lineární programování

Jedná se o nejčastěji aplikovanou oblast matematického programování. V úloze lineárního programování hledáme extrém lineární funkce více proměnných v oboru určeném lineárními rovnicemi (a/nebo lineárními nerovnicemi). Obecně to lze vyjádřit:

Nechť $a_{ij}, b_i, c_i \in \mathbf{R}$, kde $i = 1, 2, \dots, n, j = 1, 2, \dots, m; n, m \in \mathbf{N}$.

Pak minimalizujeme (popř. maximalizujeme) účelovou funkci:

$$z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

Za podmínek:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$$

...

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nm}x_n = b_n$$

$$x_i \geq 0, i = 1, \dots, n$$

Zkráceně zapisujeme ve tvaru:

$$\min_x \{ \mathbf{c}^T \mathbf{x} \mid \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$$

Pak řešení úlohy lineárního programování mohou být:

- *Přípustné řešení* je takové řešení lineárních rovnic, které vyhovuje podmínkám nezápornosti.
- *Optimální řešení* je takové přípustné řešení, které minimalizuje (resp. maximalizuje) účelovou funkci.

Každá lineární úloha může, ale nemusí mít jak přípustné tak optimální řešení. Úloha tedy může mít právě jedno optimální řešení a to v případě, že účelová funkce nabývá optimální hodnoty v krajním bodě množiny přípustných řešení, viz [15]. Úloha má nekonečně mnoho optimálních řešení, když účelová funkce nabývá optimální hodnoty ve více krajních bodech, protože dalšími optimálními řešeními jsou například body na jejich spojnicích. Další možností je, že úloha má přípustná řešení, ale neexistuje řešení optimální, protože účelová funkce na množině přípustných řešení neomezeně roste či klesá.

Jednoduché lineární úlohy se dvěma proměnnými lze řešit grafickou metodou. Pro moje modely je vhodná simplexová metoda [15], která je implementovaná v řešičích softwaru GAMA.

Modely lineárního programování umožňují řešit např. problém optimalizace výrobního programu, přiřazovací problém, minimalizaci odpadu, směšovací problém, viz [17]. Já využiji modely lineárního programování pro řešení dopravního problému a jeho modifikací.

2 Dopravní úloha

Jedná se o úlohu, kdy hledáme optimální plán přepravy výrobku od skupiny výrobců ke skupině spotřebitelů. Cílem je minimalizovat náklady na přepravu, uspokojit poptávku a nepřekročit výrobní kapacitní omezení.

Dopravní úlohu poprvé formuloval v roce 1941 americký matematik a fyzik Frank Lauren Hitchcock (1875-1957).

2.1 Formulace úlohy

Je dáno m zdrojů (dodavatelů) Z_1, \dots, Z_m s kapacitami a_1, \dots, a_m , které nám značí množství zboží, které je výrobce schopen v uvažovaném období dodat. A je dáno n cílů (spotřebitelů) C_1, \dots, C_n s požadavky b_1, \dots, b_n , což značí množství zboží, které odběratel v uvažovaném období požaduje. Množina vrcholů je pak $V = \{Z_1, \dots, Z_m, C_1, \dots, C_n\}$.

Převaha jednotkového množství od i -tého dodavatele Z_i k j -tému spotřebiteli C_j je spojena s oceněním c_{ij} , kde $ij \in E$, kde E je množina hran mezi dodavateli a spotřebiteli. Tímto oceněním mohou být myšleny náklady na přepravu jednotkového množství mezi zdrojovým a cílovým místem, nebo jejich vzájemná kilometrová vzdálenost.

Úkolem je sestavit optimální dopravní plán. To je určit hodnoty proměnných x_{ij} , kde $ij \in E$, které vyjadřují objem přepravy mezi i -tým zdrojem Z_i a j -tým cílem C_j . Tak, aby byly splněny požadavky spotřebitelů a celkové přepravní náklady z byly minimální.

Předpokládáme, že náklady c_{ij} jsou konstantní, tj. nezávisí na přepravovaném množství. Dále předpokládáme, že objem kapacit se rovná objemu požadavků, tj

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$$

Takový dopravní problém se označuje jako *vyrovnaný dopravní problém*, viz [2]. Dopravní problém, ve kterém

$$\sum_{i=1}^m a_i \neq \sum_{j=1}^n b_j$$

označujeme jako *nevyrovnaný dopravní problém*. V tom případě pak při převisu na straně nabídky zůstane část kapacity nevyužita a podobně při převisu na straně poptávky nebudou uspokojeny všechny požadavky odběratelů.

Nevyrovnaný dopravní problém lze snadno převést na vyrovnaný následujícím postupem:

- Při převisu nabídky k modelu doplníme fiktivní cílové místo C_F , jehož požadavek bude roven

$$b_F = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j$$

tj. rozdílu mezi celkovou kapacitou a celkovými požadavky.

- Při převisu poptávky k modelu doplníme fiktivní zdroj Z_F , jehož kapacita bude rovna

$$a_F = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i$$

tj. rozdílu mezi sumou požadavků a sumou kapacit.

Proto se nadále budu zabývat pouze vyrovnaným dopravním problémem.

2.2 Matematický model

Minimalizujeme náklady na přepravu:

$$z = \sum_{ij \in E} c_{ij} x_{ij} \tag{2.1}$$

při splnění omezujících podmínek.

Kapacitní omezení:

$$\sum_{j=1}^n x_{ij} \leq a_i, i = 1, \dots, m \tag{2.2}$$

Požadavky poptávky jsou dána omezeními:

$$\sum_{i=1}^m x_{ij} \geq b_j, j = 1, \dots, n \tag{2.3}$$

A hledaný tok přes hranu musí být nezáporné hodnoty:

$$x_{ij} \geq 0, \forall ij \in E \tag{2.4}$$

Chceme-li regulovat tok přes jednotlivé hrany, přidáme podmínku pro x_{ij} , kde u_{ij} je horní mez a l_{ij} dolní

$$l_{ij} \leq x_{ij} \leq u_{ij}, \forall ij \in E \tag{2.5}$$

Vlastnosti dopravního problému

- Přípustné řešení dopravního problému je vektor $x = (x_{11}, x_{12}, \dots, x_{mn})$, jehož složky splňují výše zmíněné podmínky. Množina přípustných řešení může být prázdná, omezená nebo neomezená. Viz [6].
- Po vhodné úpravě mohou být kapacity dodavatelů a požadavky odběratelů, vyjádřeny celými čísly (díky zaokrouhlování a násobení). Podle teoretických výsledků [3], [5] potom optimální řešení je celočíselné. Tato vlastnost umožňuje získat často hledaná celočíselná řešení pomocí algoritmů lineárního programování a lze se vyhnout problémům s náročností výpočtu, případně požadavku na celočíselnost optimálních hodnot proměnných.

2.3 GAMS

K modelování problému jsem použila systém GAMS (General Algebraic Modeling System), což je programovací jazyk vyšší úrovně, kterým lze formulovat matematické modely pomocí algebraických příkazů, srozumitelných jak programátorům, tak lidem zabývajících se matematickými modely, ale nečiní problémy ani praktikům s pouze částečnou znalostí obojího, viz [7].

GAMS je určen pro modelování a řešení lineárních, celočíselných a nelineárních optimalizačních problémů (popř. smíšených). Systém je uzpůsoben pro komplexní, rozsáhlé optimalizační aplikace. Umožňuje vytvořit velké udržitelné modely, které mohou být rychle přizpůsobeny novým situacím.

Byl vytvořen vývojovým týmem Světové banky v osmdesátých letech minulého století a je široce využíván na mnoha počítačových platformách. Užitečné je, že zahrnuje rozsáhlou databanku vzorových aplikačních příkladů.

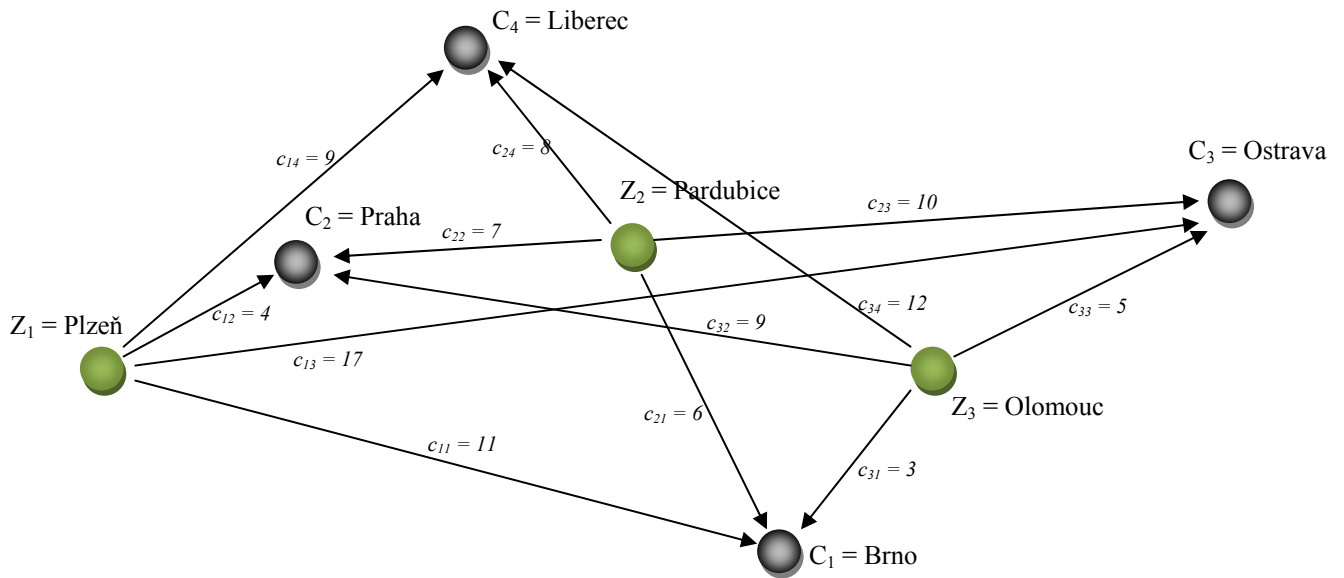
Jeho strukturu částečně vysvětlím na příkladech 2.1 a 2.2.

2.4 Příklad 2.1

[2] Společnost Multicomp, s.r.o. má v České republice tři střediska (Plzeň, Pardubice, Olomouc), ve kterých montuje osobní počítače. Kapacita těchto středisek je 330, 150 a 220 kusů počítačů měsíčně. Tyto počítače jsou distribuovány smluvním odběratelům v Brně, Praze, Ostravě a Liberci. Podle smluv dodá Multicomp jednotlivým odběratelům postupně 180, 250, 160 a 110 kusů počítačů. Distribuční náklady mezi středisky a odběrateli jsou vykalkulovány na 1ks počítače ve výši, která je uvedena v matici C (uvedené hodnoty jsou ve stovkách Kč).

$$C = \begin{pmatrix} - & \text{Brno} & \text{Praha} & \text{Ostrava} & \text{Liberec} \\ \text{Plzeň} & 11 & 4 & 17 & 9 \\ \text{Pardubice} & 6 & 7 & 10 & 8 \\ \text{Olomouc} & 3 & 9 & 5 & 12 \end{pmatrix}$$

Příklad můžeme zobrazit pomocí teorie grafů – viz *Obr.2.1*. Dostaneme tak jednoduchý orientovaný ohodnocený graf.



Obr. 2.1 – Zobrazení příkladu na grafu

Pak matematická formulace daného příkladu je následující:

Minimalizovat

$$z = 11x_{11} + 4x_{12} + 17x_{13} + 9x_{14} + 6x_{21} + 7x_{22} + 10x_{23} + 8x_{24} + 3x_{31} + 9x_{32} + 5x_{33} + 12x_{34}$$

za podmínek

$$x_{11} + x_{12} + x_{13} + x_{14} \leq 330$$

$$x_{21} + x_{22} + x_{23} + x_{24} \leq 150$$

$$x_{31} + x_{32} + x_{33} + x_{34} \leq 220$$

$$x_{11} + x_{21} + x_{31} \geq 180$$

$$x_{12} + x_{22} + x_{32} \geq 250$$

$$x_{13} + x_{23} + x_{33} \geq 160$$

$$x_{14} + x_{24} + x_{34} \geq 110$$

$$x_{ij} \geq 0, i = 1, 2, 3, j = 1, 2, 3, 4$$

Při programování v GAMSu budu problém modelovat s užitím matice sousednosti.

Implementace daného modelu v GAMSu:

Sets

```
i zdroj / Plzen, Pardubice, Olomouc /
j cil / Brno, Praha, Ostrava, Liberec / ;
```

Příkaz *Set* slouží k definici množin (určených většinou výčtem indexů) používaných v modelu. V našem případě vytvoříme dvě množiny *i* a *j*. Přičemž *i* (resp. *j*) je název množiny a *zdroj* (resp. *cil*) je vysvětlující popis pro uživatele. Tento popis se pak přenáší do všech výstupů vztahující se k množině *i* (resp. *j*). To vede k značné přehlednosti výsledků. V lomítkových závorkách je pak uveden výčet prvků množiny. Středníkem je pak označen konec příkazu.

Parameters

```
a(i) kapacita zdroje
    /   Plzen      330
      Pardubice  150
      Olomouc    220 /

b(j) požadavek cile
    /   Brno      180
      Praha     250
      Ostrava    160
      Liberec    110 / ;
```

Příkaz *Parameters* se užívá k zadávání jednorozměrných (někdy i vícerozměrných) polí. Název parametru je *a*, *i* určuje množinu indexů, pro kterou se hodnota parametru mění. Přiřazení hodnot parametru indexů je opět v lomítkových závorkách. Jako oddělovače slouží čárky (jako výše) nebo konce řádků. Nedefinované prvky se automaticky berou jako nulové.

Table *c(i,j)* distribucni naklady mezi uzly

	Brno	Praha	Ostrava	Liberec
Plzen	11	4	17	9
Pardubice	6	7	10	8
Olomouc	3	9	5	12 ;

Příkaz *Table* definuje hodnoty dvou a vícerozměrných parametrů. Opět je uveden název tabulky a v závorkách jsou uvedeny množiny, jejichž prvky je tabulka indexována. Při přiřazování konkrétních hodnot uvedeme do prvního řádku výpis prvků z množiny *j* a do prvního sloupce výpis prvků z množiny *i*.

Variable

```
z celkove naklady ;
```

Positive Variable

```
x(i,j) objem prepravy mezi i-tym a j-tym uzlem ;
```

Deklaraci proměnných uvozuje klíčové slovo *Variable*. Použijeme-li *Positive Variable*, určujeme obor hodnot deklarované proměnné na nezáporné. Podobně můžeme deklarovat proměnné typu *Negative* (nekladný obor hodnot), *Binary* (hodnoty 0 nebo 1) nebo *Integer* (celočíselné hodnoty od 0 do 100).

Každá proměnná označuje záznam, který se skládá ze čtyř hodnot, které se označují pomocí „přípon“ k označení proměnné: *.lo*, *.up*, *.l*, *.m*. První z hodnot *.lo* určuje dolní mez pro hodnoty proměnné, *.up* určuje horní hranici, *.l* určuje vlastní optimální hodnotu proměnné a *.m* její marginální (duální) hodnotu, viz [7], [8].

Equations

```
ucelfce, omez1(i), omez2(j);

ucelfce .. z =e= sum((i,j), c(i,j)*x(i,j)) ;
```

```
omez1(i) .. sum(j, x(i,j)) =l= a(i) ;
omez2(j) .. sum(i, x(i,j)) =g= b(j) ;
```

Klíčové slovo *Equations* označuje deklaraci rovnic a nerovnic definujících množinu přípustných řešení modelu a účelovou funkci. V našem případě to jsou rovnice: *ucelfce* ... (2.1), nerovnice *omez1(i)* ... (2.2), nerovnice *omez2(j)* ... (2.3).

Nejprve se uvádí název omezení, které definujeme. Jako oddělení od jeho zápisu se používají dvě tečky. Pak následuje vlastní definitorský zápis. Přičemž rovnost a nerovnost se značí následujícími symboly: $= l =$ (ve smyslu menší nebo rovno), $= g =$ (větší nebo rovno), $= e =$ (rovno). V samotném předpisu se mohou užívat standardní operace a funkce. Taktéž i indexové funkce jako sumace (viz zdrojový kód výše : *sum*).

Rovnice se nemusí nijak upravovat. GAMS provede sám převod proměnných na jednu stranu.

```
Model transport /all/ ;
```

Po klíčovém slově *Model* následuje název modelu a v lomítkových závorkách jsou uvedeny rovnice (nerovnice), které mají být v modelu zahrnuty. Tedy můžeme definovat více rovnic (nerovnic) a pak použít jen některé a také vytvořit více modelů. Pokud chceme zahrnout do modelu všechny rovnice (nerovnice) použijeme klíčové slovo *all*.

```
Solve transport using lp minimizing z ;
```

Na spuštění vlastního řešení modelu je příkaz *Solve*. Po tomto klíčovém slově následuje název modelu. Příkazem *minimizing* nebo *maximizing* určujeme, zda se jedná o minimalizační či maximalizační úlohu. *Using* uvozuje typ úlohy. Ten může být následující:

- LP – úloha lineárního programování
- NLP – úloha nelineárního programování
- MIP – úloha celočíselného programování (RMIP – relaxovaná úloha celočíselného programování, požadavek celočíselnosti proměnných nemusí být splněn)
- MINLP – kombinace MIP a NLP

```
Display x.l, z.l ;
```

Na závěr zobrazíme výsledné hodnoty proměnné x a minimalizované celkové náklady z pomocí příkazu *Display*. Což je naše hledané optimální řešení. Vypisujeme jejich vlastní hodnotu značenou tečkovým zápisem $.l$.

```

----      46 VARIABLE x.L  objem prepravy mezi i-tym a j-tym uzlem

                Brno      Praha      Ostrava      Liberec

Plzen                250.000                80.000
Pardubice      120.000                30.000
Olomouc        60.000                160.000

----      46 VARIABLE z.L                =      3660.000  celkove naklady

```

Ve výpisu výsledků GAMSu je uvedeno optimální řešení v tabulce, tzn. že např. z Plzně do Prahy musí být přepraveno 250 jednotek a celkové náklady na dopravu jsou rovny 3660.

2.5 Příklad 2.2

Vycházíme z modelu příkladu 2.1, ovšem počet zdrojů a cílů bude omezen pouze výkonností výpočetní jednotky. Toto omezení jsem na závěr testovala pro svůj osobní počítač. Časově jsem se pohybovala mezi sekundami až 10 minutami, tj 100×100 proměnných až 2200×2200 proměnných. V okamžiku kdy jsem překročila počtem dat 2200×2200 , došlo k přetečení paměti. Naměřené hodnoty viz tabulka *Tab. 2.1* a její grafické znázornění na *Obr.2.2*.

Kapacity a požadavky budou náhodně generovány z množiny $\{0, 1, 2, \dots, 1000\}$ a distribuční náklady mezi uzly z množiny $\{1, 2, \dots, 100\}$.

Implementace v GAMSu:

Opět máme dvě množiny indexů zdrojů a cílů. Ovšem u obou máme přidány tzv. fiktivní zdroj, resp. cíl, f . Neboť bereme náhodně vygenerovaná čísla, nemůžeme předpokládat, že půjde o vyrovnaný model.

Sets

```

i zdroj / 1*999, f /
j cil / 1*999, f / ;

```

Parameters

```

a(i) kapacita zdroje
b(j) požadavek cile
c(i,j) distribucni naklady mezi uzly ;

```

Náhodně vygenerujeme parametry a, b, c z celých čísel:

```

a(i) = uniformint (0,1000) ;
b(j) = uniformint (0,1000) ;
c(i,j) = uniformint (1,100) ;

```

Vynulujeme hodnoty vázající se k fiktivním uzlům:

```

a('f')=0; b('f')=0;
c(i,'f')=0; c('f',j)=0;

```

Metodou popsanou výše zjistíme vyváženost – nevyváženost modelu a popřípadě jej převedeme na model vyvážený.

```

if ( sum(i, a(i)) <> sum(j, b(j)),
    if ( sum(i, a(i)) < sum(j, b(j)),
        a('f') = sum(j, b(j))-sum(i, a(i));
    else b('f') = sum(i, a(i))-sum(j, b(j));
    );
);

```

Dále se postupuje obdobně jako u Příkladu 2.1.

Na závěr se můžeme pomocí příkazu PUT podívat na data, která nás zajímají. Například se můžeme podívat na množství zboží odcházející od dodavatele 8, a do kterého cíle putuje.

```

File OUT / 'DopProbVD.TXT' /;

Put OUT;
Put 'Vysledky:' /;
Put '======' / /;
Loop(j,
    If(x.l('8',j)>0,
        Put 'Odchozí zboží ze zdroje Z8: ', x.l('8',j);
        Put '          do cile:      ', j.tl:5 /;
    );
);

```

Výstupem je pak textový soubor *DopProbVD.txt*, kam se požadovaná data uloží.

Vysledky:
=====

```

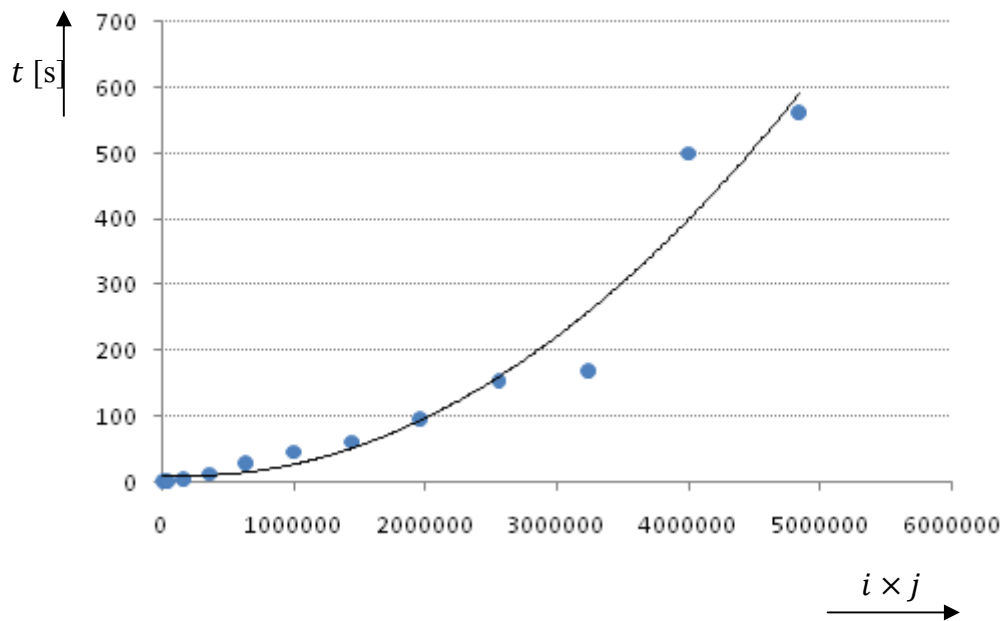
Odchozí zboží ze zdroje Z8:      291.00      do cile:      1
Odchozí zboží ze zdroje Z8:      122.00      do cile:      3
Odchozí zboží ze zdroje Z8:      150.00      do cile:      4
Odchozí zboží ze zdroje Z8:      231.00      do cile:      7
Odchozí zboží ze zdroje Z8:       63.00      do cile:     14

```

Závěrem uvedu ilustraci růstu potřebného času k výpočtu na velikosti úlohy.

Tab. 2.1 - Tabulka výsledků pokusu

počet zdrojů <i>i</i>	počet cílů <i>j</i>	čas <i>t</i> [m:s]
100	100	0:00,344
200	200	0:01,203
400	400	0:04,828
600	600	0:11,968
800	800	0:27,953
1000	1000	0:45,484
1200	1200	1:00,828
1400	1400	1:34,703
1600	1600	2:33,250
1800	1800	3:08,610
2000	2000	8:19,453
2200	2200	9:22,625



Obr. 2.2 – Závislost času výpočtu na počtu uzlů

Jak můžeme vidět na *Obr 2.2*, znázorněné hodnoty naznačují zrychlení růstu časové náročnosti výpočtu pro větší úlohy.

3 Přepravní úloha

Jedná se o zobecněný případ dopravního problému. V těchto úlohách je dodávka přepravována ze zdroje do cíle přes tzv. přepravní uzly, viz [15].

3.1 Formulace speciální úlohy

Je dáno m zdrojů (dodavatelů) Z_1, \dots, Z_m s kapacitami a_1, \dots, a_m , které nám značí množství zboží, které je výrobce schopen v uvažovaném období dodat. A je dáno n cílů (spotřebitelů) C_1, \dots, C_n s požadavky b_1, \dots, b_n , což značí množství zboží, které odběratel v uvažovaném období požaduje. Taktéž je dáno p přepravních uzlů P_1, \dots, P_p , pro něž definujeme kapacity (resp. požadavky) rovny 0.

Přeprava jednotkového množství od dodavatele ke spotřebiteli je opět spojena s oceněními c_{ij} , kde $ij \in E$, kde E je množina hran mezi některými vrcholy indexovanými i a j , určená následovně. $V = \{Z_1, \dots, Z_m, C_1, \dots, C_n, P_1, \dots, P_p\}$ je množina všech vrcholů. Od každého dodavatele vede hrana do přepravního uzlu a z každého přepravního uzlu vede hrana ke každému spotřebiteli.

Naším úkolem je sestavit optimální dopravní plán. To je určit hodnoty proměnných x_{ij} tak, aby byly splněny požadavky spotřebitelů a celkové přepravní náklady z byly minimální.

Předpokládáme, že náklady c_{ij} jsou konstantní, tj. nezávisí na přepravovaném množství. Dále předpokládáme vyrovnaný model.

3.2 Matematický model

Minimalizujeme náklady na přepravu: (3.1)

$$z = \sum_{ij \in E} c_{ij} x_{ij}$$

při splnění omezujících podmínek.

Kapacitní omezení: (3.2)

$$\sum_{j=1}^n x_{ij} \leq a_i, i = 1, \dots, m$$

Požadavky poptávky jsou dána omezeními:

(3.3)

$$\sum_{i=1}^m x_{ij} \geq b_j, j = 1, \dots, n$$

Omezení pro přepravní uzel: (3.4)

$$\sum_{i=1}^m x_{ik} = \sum_{j=1}^n x_{kj}, k = 1, \dots, p$$

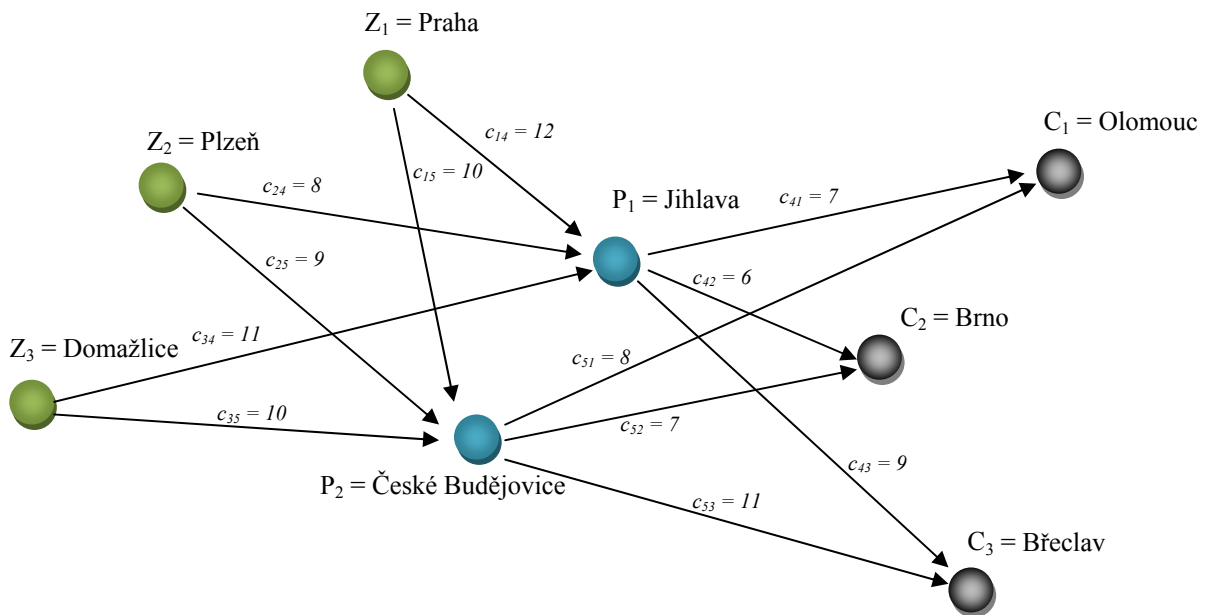
A hledané množství zboží musí být nezáporné hodnoty: (3.5)

$$x_{ij} \geq 0, \forall ij \in E$$

3.3 Příklad 3.1 – Speciální přepravní problém

Speciální přepravní problém spočívá v tom, že neexistují přímé hrany mezi zdroji a cíly. Vždy se musí jít přes přepravní uzel.

Společnost Fiktiv, s.r.o. má v České republice tři výroby a to v Praze, Plzni a Domažlicích, ve kterých produkuje paměťové karty do fotoaparátů. Kapacita těchto továren je 50, 70 a 80 kusů karet měsíčně. Tyto karty jsou pak distribuovány smluvním odběratelům v Olomouci, Brně a Břeclavi. Podle smluv dodá Fiktiv jednotlivým odběratelům 70, 60 a 70 karet. Přičemž distribuce probíhá přes přepravní uzly, které jsou v Jihlavě a v Českých Budějovicích. Distribuční náklady mezi jednotlivými uzly jsou vykalkulovány na 1ks počítače ve výši, která je uvedena v grafu – viz *Obr. 3.1*.



Obr. 3.1 – Zobrazení příkladu na grafu

Pak matematická formulace daného příkladu je následující:

Minimalizovat

$$z = 12x_{14} + 10x_{15} + 8x_{24} + 9x_{25} + 11x_{34} + 10x_{35} + 7x_{41} + 6x_{42} + 9x_{43} + 8x_{51} + 7x_{52} + 11x_{53}$$

za podmínek (protože se jedná o vyrovnaný model, uvádíme rovnosti)

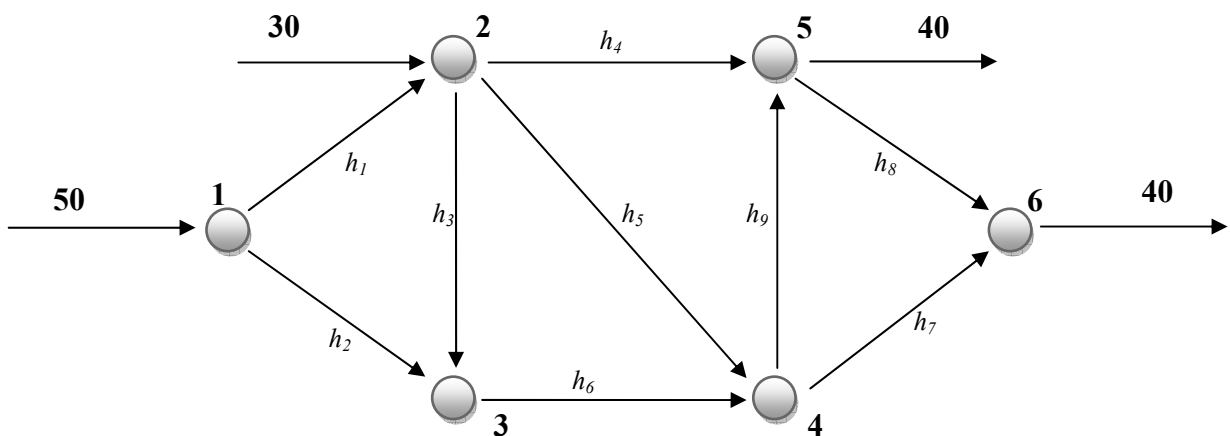
$$x_{14} + x_{15} = 50$$

$$\begin{aligned}
x_{24} + x_{25} &= 70 \\
x_{34} + x_{35} &= 80 \\
x_{41} + x_{51} &= 70 \\
x_{42} + x_{52} &= 60 \\
x_{43} + x_{53} &= 70 \\
x_{14} + x_{24} + x_{34} &= x_{41} + x_{42} + x_{43} \\
x_{15} + x_{25} + x_{35} &= x_{51} + x_{52} + x_{53} \\
x_{ij} &\geq 0, i = 1, 2, 3, 4, 5, j = 1, \dots, 5
\end{aligned}$$

Uvedený model jde implementovat jako předcházející dopravní problém. My však zvolíme obecnější postup pomocí incidenční matice.

3.4 Obecný přepravní problém

V obecné přepravní úloze mohou být hrany mezi libovolnými uzly. Tedy i přímo mezi zdrojem a cílem nebo mezi přepravními uzly – viz Obr. 3.2



Obr. 3.2 – Graf obecného přepravního problému

Obecné přepravní úlohy lze modelovat pomocí toků v sítích (např. přepravní silniční síť, viz [15]), což vede na úlohu lineárního programování

$$\min_x \{ \mathbf{c}^T \mathbf{x} \mid \mathbf{A} \mathbf{x} = \mathbf{b}, \mathbf{x} \geq \mathbf{0} \}$$

s proměnnými a parametry:

\mathbf{c} ... je vektor se složkami c_{ij}

c_{ij} ... cena jednotky toku skrz hranu ij , tj. spojnice bodů i a j

\mathbf{x} ... je vektor se složkami x_{ij}

x_{ij} ... tok přes hranu ij , tj. přepravené množství mezi body i a j

\mathbf{A} ... incidenční matice grafu, kde řádky odpovídají uzlům a sloupce hranám

\mathbf{b} ... vektor poptávky, kde b_i pro uzel i představuje přítok nebo odtok do uzlu i

Motivační příklad

Zavádíme znaménkovou konvenci: pro přicházející hranu + a pro odchozí hranu −.

Mějme tedy graf *Obr. 3.2*.

Pak můžeme sestavit incidenční matici A pro daný graf

$$A = \begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & -1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Minimalizujeme účelovou funkci přes x :

$$z = x_{12} + 2x_{13} + 3x_{23} + 4x_{25} + 5x_{24} + 6x_{34} + 7x_{46} + 8x_{56} + 9x_{45}$$

Za podmínek:

$$-x_{12} - x_{13} = -50$$

$$x_{12} - x_{23} - x_{24} - x_{25} = -30$$

$$x_{13} + x_{23} - x_{34} = 0$$

$$x_{24} + x_{34} - x_{46} - x_{45} = 0$$

$$x_{25} - x_{56} + x_{45} = 40$$

$$x_{46} + x_{56} = 40$$

$$x_{ij} \geq 0, ij \in E, \text{ kde } E \text{ je množina všech hran grafu na } Obr. 3.2$$

Tento model podobně jako předcházející jde implementovat v GAMSu.

3.5 Implementace příkladu 3.1 v GAMSu

Jak obecnou, tak speciální přepravní úlohu lze zapsat pomocí incidence jako úlohu lineárního programování. Toho nyní využijeme při implementaci a řešení příkladu 3.1.

Mějme tedy dvě množiny: V značí množinu všech uzlů grafu, E je pak množina orientovaných hran mezi vrcholy $u, v \in V$.

Sets

```
i  uzly      / Z1, Z2, Z3, P1, P2, C1, C2, C3 /
j  hrany     / Z1-P1, Z1-P2, Z2-P1, Z2-P2, Z3-P1, Z3-P2,
              P1-C1, P1-C2, P1-C3, P2-C1, P2-C2, P3-C3 / ;
```

Jednotlivé kapacity a požadavky uzlů zapíšeme jako parametr $b(i)$. Záporné hodnoty značí kapacity zdrojů, kladné hodnoty požadavky cílů. Přepravní uzly mají kapacitu 0.

Parameters

```
b(i)  kapacita uzlu    /  Z1  -50
                                Z2  -70
                                Z3  -80
                                P1   0
                                P2   0
                                C1   70
                                C2   60
                                C3   70  /
```

Pro graf na *obr.3.1* sestavíme incidenční matici $A(i,j)$.

Table $A(i,j)$ **incidenční matice grafu**

```
          Z1-P1  Z1-P2  Z2-P1  Z2-P2  Z3-P1  Z3-P2
Z1      -1      -1
Z2              -1      -1
Z3              -1      -1
P1       1              1
P2              1              1
C1
C2
C3

+

          P1-C1  P1-C2  P1-C3  P2-C1  P2-C2  P3-C3
Z1
Z2
Z3
P1      -1      -1      -1
P2              -1      -1      -1
C1       1              1
C2              1              1
C3              1              1 ;
```

Kterou pak použijeme v kapacitním omezení (nejprve je uvedena účelová funkce)

Equations

```
ucelfce, omez1(i);

ucelfce ..      z  =e= sum((j), c(j)*x(j)) ;

omez1(i) ..      sum(j, A(i,j)*x(j))  =e=  b(i) ;
```

A dostáváme optimální přepravu mezi jednotlivými uzly a minimalizované celkové náklady.

```
----      72 VARIABLE x.L  objem prepravy mezi i-tym a j-tym uzlem

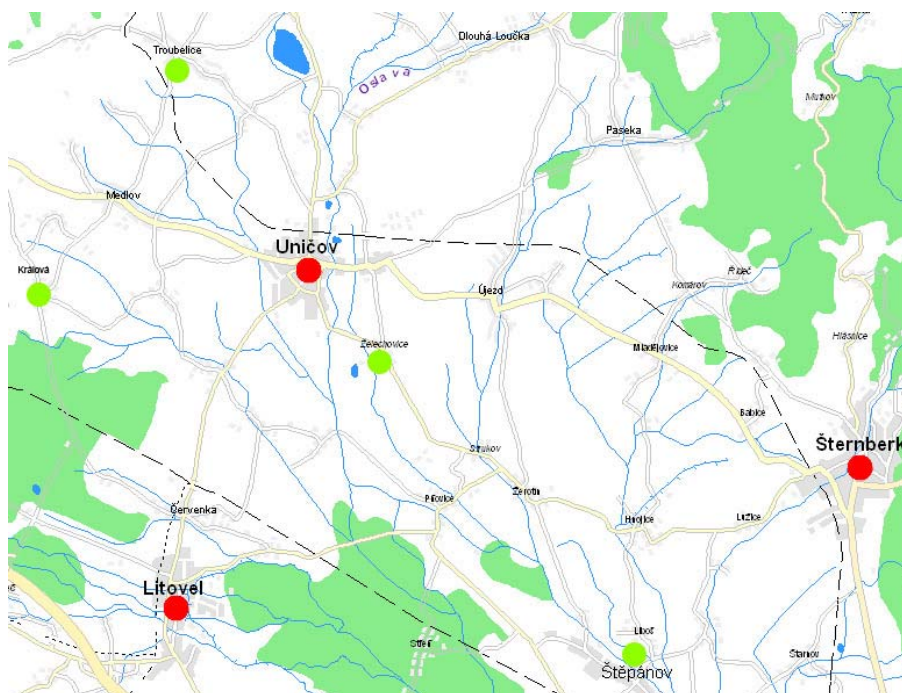
Z1-P2 50.000,      Z2-P1 70.000,      Z3-P1 80.000,      P1-C1 20.000,
P1-C3 70.000,      P2-C1 50.000,      P1-C2 60.000

----      72 VARIABLE z.L          =      3470.000  celkove naklady
```

4 Aplikační příklad

Na Olomoucku sídlí firma Skopal zabývající se přepravou levné pracovní síly z vesnic s nástupem v Troubelicích, Králové, Želechovicích a ve Štěpánově, do blízkých měst Uničova, Litovle a Šternberka. Viz mapa *Obr. 4.1*. Zdroj map [14].

Kapacita sběrných míst postupně: 17, 10, 13 a 20 lidí denně. Poptávka ve městech je 20, 13 a 27 lidí denně na veřejně prospěšné práce, úklid města, pomocné práce při stavbě, atd. Tedy po lidech bez nutné kvalifikace a odborných znalostí.



Obr. 4.1 – mapa okolí města Uničova, kde jsou červenou vyznačeny cíle a zelenou zdroje

Označení vrcholů:

- 1...Troubelice
- 2...Králová
- 3...Želechovice
- 4...Štěpánov
- 5...Uničov
- 6...Litovel
- 7...Šternberk

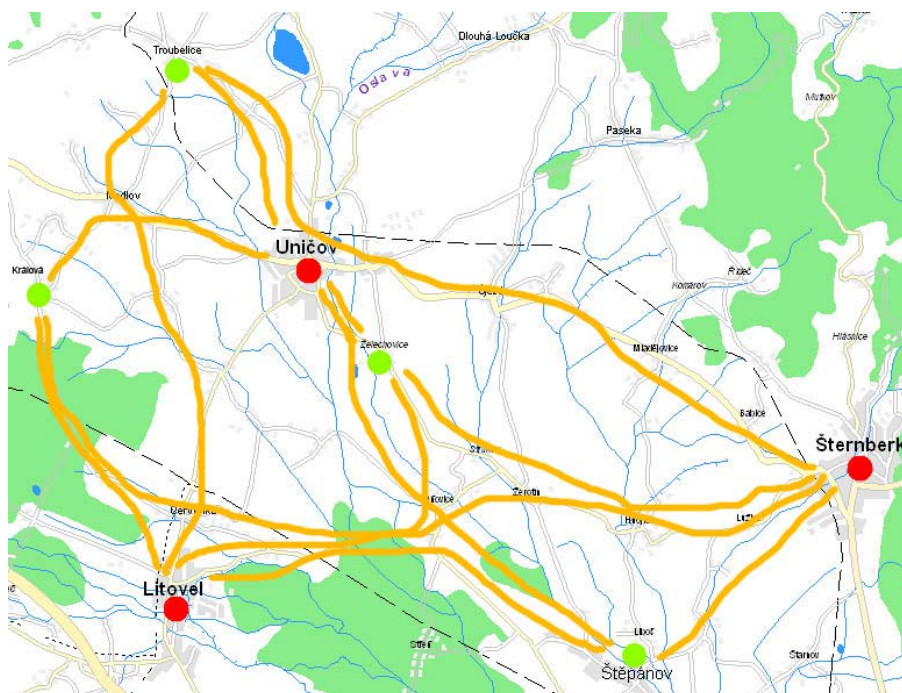
4.1 Aplikace dopravního modelu

Uvažujeme rozmístění hran naznačených na mapě *Obr. 4.2*, kterou můžeme převést na graf *Obr. 4.3*.

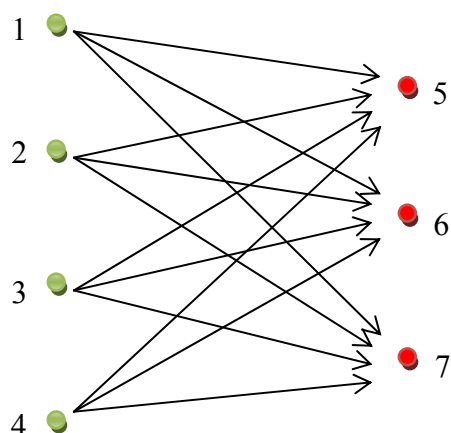
Přepravní náklady mezi uzly uvažujeme v kilometrové vzdálenosti krát 10, které jsou uvedeny v tabulce *Tab. 4.1*.

Tab. 4.1. – Přepravní náklady

	<i>Troubelice</i>	<i>Králová</i>	<i>Želechovice</i>	<i>Štěpánov</i>
<i>Uničov</i>	63	78	35	166
<i>Litovel</i>	151	93	83	149
<i>Šternberk</i>	215	230	177	90



Obr. 4.2 – Naznačení hran



Obr. 4.3 – Převodění mapy na graf

Implementace v GAMSu

\$Title: Dopravní problém

Sets

```
i  uzly      / 1*7 /
j  hrany     / 1-5, 1-6, 1-7, 2-5, 2-6, 2-7,
              3-5, 3-6, 3-7, 4-5, 4-6, 4-7 /;
```

Parameters

```
c(j)  distribucni naklady pres hranu j
/ 1-5 63, 1-6 151, 1-7 215, 2-5 78, 2-6 93, 2-7 230,
  3-5 35, 3-6 83, 3-7 177, 4-5 166, 4-6 149, 4-7 90 /
b(i)  kapacita uzlu i
/1 -17, 2 -10, 3 -13, 4 -20, 5 20, 6 13, 7 27 / ;
```

Table A(i,j) incidencni matice grafu

```
      1-5 1-6 1-7 2-5 2-6 2-7 3-5 3-6 3-7 4-5 4-6 4-7
1      -1 -1 -1
2              -1 -1 -1
3                      -1 -1 -1
4                                -1 -1 -1
5      1              1              1
6              1              1              1
7              1              1              1 ;
```

Variable

```
z      min naklady ;
```

Positive Variable

```
x(j)      objem prepravy pres hranu j ;
```

Equations

```
ucelfce, omez1(i);
```

```
ucelfce .. z =e= sum((j), c(j)*x(j)) ;
```

```
omez1(i) .. sum(j, A(i,j)*x(j)) =e= b(i) ;
```

```
Model transport /all/ ;
```

```
Solve transport using lp minimizing z ;
```

```
Display x.l, z.l ;
```

Optimální řešení

Výstup v GAMSu:

```
----      42 VARIABLE x.L  objem prepravy pres hranu j

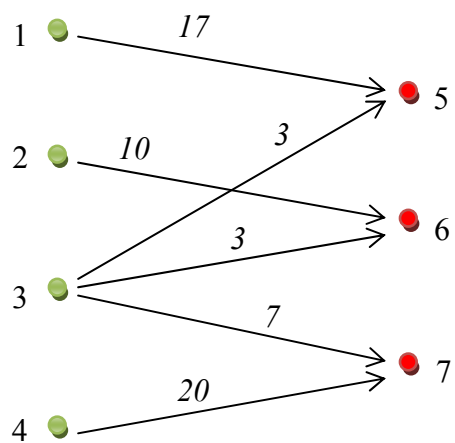
1-5 17.000,    2-6 10.000,    3-5  3.000,    3-6  3.000,    3-7  7.000
4-7 20.000

----      42 VARIABLE z.L              =      5394.000  min naklady
```

Pro přehlednost převedeme optimální přepravu do tabulky *Tab. 4.2* a graficky zobrazíme v grafu *Obr. 4.4*. Celkové minimalizované náklady jsou $z = 5394$.

Tab. 4.2 – Přeprava mezi uzly

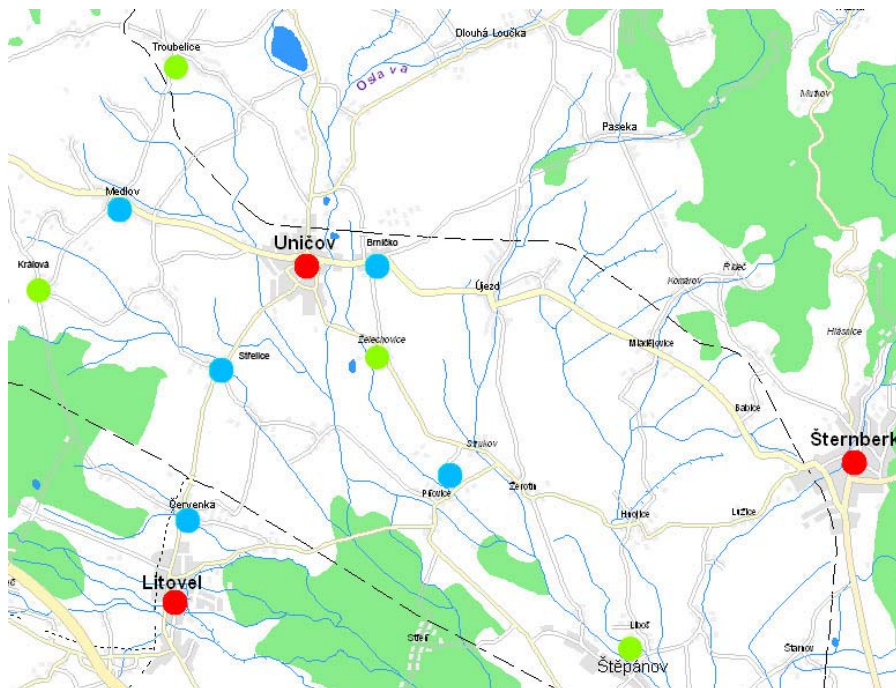
	Uničov	Litovel	Šternberk
Troubelice	17		
Králová		10	
Želechovice	3	3	7
Štěpánov			20



Obr. 4.4 – Výsledné zobrazení toků

4.2 Aplikace přepravního modelu

Přepravní problém vznikne přidáním k vrcholům 1, ..., 7 přepravních uzlů 8, ..., 12. Viz mapa *Obr. 4.5*.

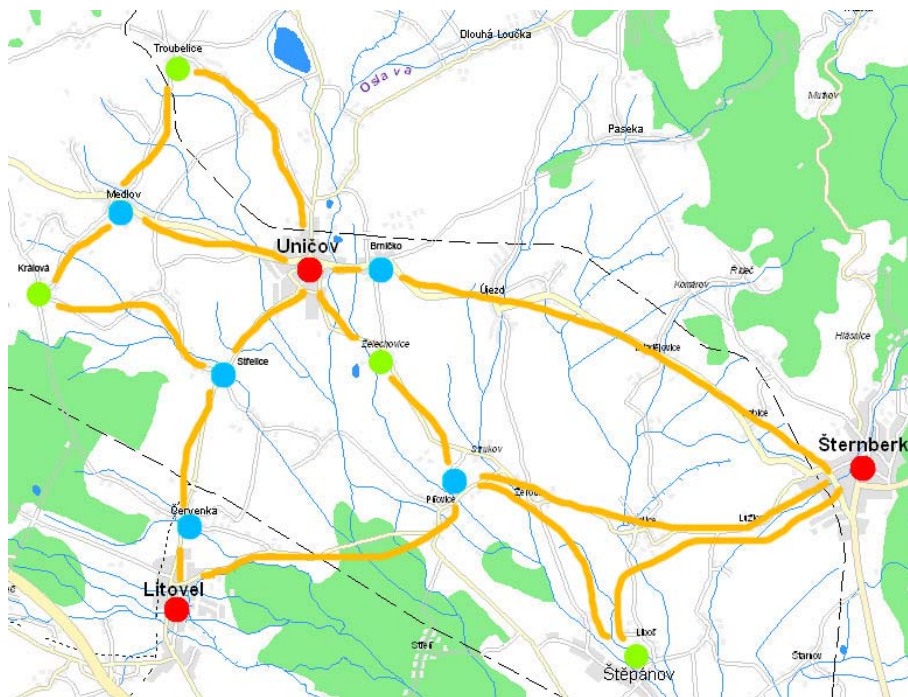


Obr. 4.5 – mapa okolí města Uničova, kde jsou červenou vyznačeny cíle, zelenou zdroje a modrou přepravní uzly

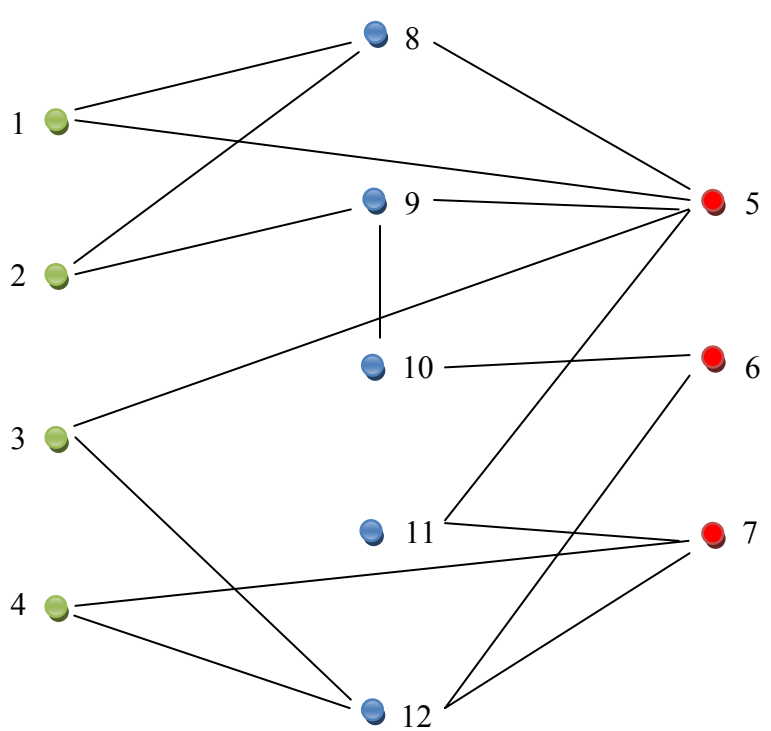
Označení přepravních uzlů:

- 8...Medlov
- 9...Střelice
- 10...Červenka
- 11...Brničko
- 12...Pňovice

Naznačení hran viz *Obr.4.6* a jeho převedení do grafu viz *Obr. 4.7*. Přepravní náklady mezi uzly uvažujeme v kilometrové vzdálenosti krát 10, které jsou uvedeny v tabulce *Tab.4.3*. Uvedení přepravních uzlů umožňuje možnost přesedání.



Obr. 4.6 – Naznačení hran



Obr. 4.7 – Převedení mapy na graf, přičemž po hraně se můžeme pohybovat oběma směry

Tab. 4.3. – Převážní náklady

	1	2	3	4	5	6	7	8	9	10	11	12
1					63			38				
2								33	57			
3					35							59
4							90					72
5	63		35					45	39		15	
6										23		77
7				90							137	118
8	38	33			45							
9		57			39					41		
10						23			41			
11					15		77					
12			59	72		77	118					

Optimální řešení

Výstup v GAMSu:

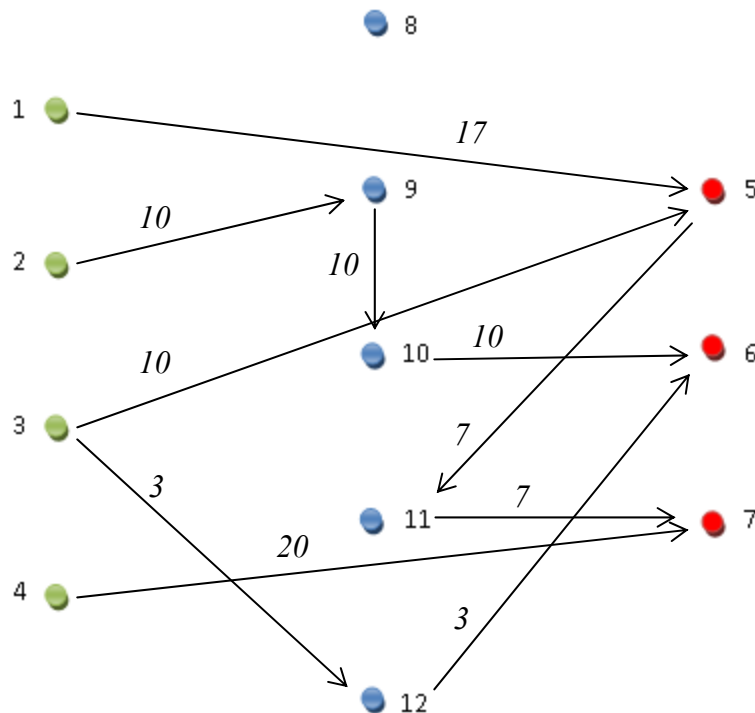
```
----      67 VARIABLE x.L   objem prepravy pres hranu j

1-5 17.000,    2-9 10.000,    3-5 10.000,    3-12 3.000,    4-7 20.000
5-11 7.000,    9-10 10.000,    10-6 10.000,    11-7 7.000,    12-6 3.000

----      67 VARIABLE z.L           =      5483.000   min naklady
```

Pro přehlednost jsou toky znázorněny do grafu *Obr. 4.8*. Celkové minimalizované náklady jsou $z = 5483$.

Pokud porovnáme vstupní informace u dopravního problému 4.1 a přepravního problému, můžeme přepravní problém považovat za zpřesnění původní dopravní úlohy. Neboť u dopravní úlohy vzdálenosti mezi městy byly odhadnuty bez uvažování jemnější přepravní sítě zahrnující přepravní uzly. Díky tomuto postupu byly tyto odhady vzdáleností nižší než při uvažování skutečných vzdáleností s přepravními uzly. Proto výslednou optimální hodnotu účelové funkce u dopravní úlohy lze považovat za dolní mez přesněji určené optimální hodnoty účelové funkce u přepravní úlohy. Jedná se vlastně o důsledek, že v přepravní úloze hraje svoji roli trojúhelníková nerovnost, která nám zde konkrétně říká, že cesta přes přepravní uzel je delší nebo rovna než přímá vzdálenost.



Obr. 4.8 – Výsledné toky v síti

5 Problém náhodnosti poptávky

Obecný přepravní problém lze rozšířit o náhodnost poptávky. Při nestálé poptávce, kdy dochází k jejím nečekaným změnám, se musí dodavatel rozhodnout, kolik zboží pošle. Tomuto rozhodování v podmínkách neurčitosti může být nápomocno matematické programování – přesněji stochastické programování.

5.1 Stochastické programování

Stochastické programování se začalo rozvíjet v padesátých letech minulého století. Jednalo se o problémy zahrnující nejistotu a nutnost rozhodnout v daném časovém období. První teoretické výsledky byly zveřejněny v šedesátých letech. Viz [10].

Mnoho životních situací zahrnuje nejistotu a náhodnost. Viz např: ve výrobě – nejisté náklady, v dopravě – nejisté požadavky, ve správě lesů – nepředpovídatelnost požárů, vichřic, atd.

Náhodnost se většinou značí symbolem ξ , zápis $\mathbf{b}(\xi)$ značí, že vektor \mathbf{b} má náhodné elementy.

Úlohu stochastického programování lze obecně formulovat v následujícím tvaru:

$$\min_x \{f(\mathbf{x}, \xi) | \mathbf{x} \in C(\xi) = \{\mathbf{x} \in X \subseteq \mathbf{R}^n | g(\mathbf{x}, \xi) \circ 0\}\}$$

Tedy dodáme-li náhodnost poptávky $\mathbf{b}(\xi)$ do našeho modelu, dostaneme:

$$\min_x \{c^T \mathbf{x} | A\mathbf{x} = \mathbf{b}(\xi), \mathbf{x} \geq \mathbf{0}\}$$

5.2 „Wait – and – see“ přístup (WS)

Hlavní problém je, ve kterém časovém okamžiku budeme moci rozhodnout. Povaha aplikačního problému obvykle určí, že se můžeme rozhodnout buď předtím, než je náhodný parametr ξ pozorován nebo až po pozorování, kdy už je jeho hodnota známa.

Přístup WS značí ten druhý případ – před realizací rozhodnutí známe realizaci náhodného parametru ξ , tedy naše rozhodnutí \mathbf{x} může využít tuto informaci. Modifikujeme tedy naše rozhodnutí podle pozorování a z toho důvodu můžeme rozhodnutí považovat za funkci náhodného parametru, zapisujeme $\mathbf{x}(\xi)$. Pak také hodnota účelové funkce je náhodná proměnná.

Tento přístup má svůj význam v aplikacích matematických statistiky.

5.3 „Here – and – now“ přístup (HN)

Rozhodnutí \mathbf{x} musí být provedeno před v současnosti neznámou budoucí realizací náhodného parametru ξ . Při řešení našeho aplikačního příkladu budeme vycházet z toho, že naše rozhodnutí musí být provedeno, než je poptávka známa.

5.4 Scénářové úlohy

Nechť ξ je diskrétní náhodný vektor na pravděpodobnostním prostoru s konečným počtem možných realizací r , $r \in N_0$. Množina $S = \{1, 2, \dots, r\}$.

Symbolem p_s označíme pravděpodobnost, že nastane daná realizace ξ^s , $\forall s \in S$. Z vlastností pravděpodobnostní funkce plyne

$$\sum_{s \in S} P(\xi = \xi^s) = \sum_{s \in S} p_s = 1$$

Nadále budeme uvažovat pouze ty realizace ξ^s , jejichž pravděpodobnost p_s je nenulová. Tyto realizace se nazývají scénáře.

5.5 Formulace úlohy

Jelikož se potýkáme s náhodnou poptávkou a naše rozhodnutí je HN, může nastat situace, kdy je zboží nedostatek či přebytek. To řešíme tzv. kompenzačním rozhodnutím y_{is} příslušnému danému vrcholu i , $i \in I \subset V$, kde množina I obsahuje všechny cílové vrcholy, a scénáři $s \in S$. Tedy v případě nedostatečné poptávky musíme zboží y_{is}^- odebrat za cenu q_i^- . A naopak při neuspokojení poptávky musíme na naše další náklady q_i^+ dopravit chybějící zboží y_{is}^+ . Přirozeně se snažíme dodatečné náklady minimalizovat.

Taktéž musíme zahrnout pravděpodobnost p_s , $s \in S$, každého možného scénáře, což učiníme tak, že budeme minimalizovat střední hodnotu dodatečných nákladů.

5.6 Matematický model

Minimalizujeme

$$z = \sum_{ij \in E} c_{ij} x_{ij} + \sum_{s \in S} p_s \left(\sum_{i \in I} q_i^+ y_{is}^+ + \sum_{i \in I} q_i^- y_{is}^- \right) \quad (5.1)$$

Za podmínek

$$\sum_{jk \in E} a_{i(jk)} x_{jk} + y_{is}^+ - y_{is}^- = b_{is}, \forall s \in S, \forall i \in I \quad (5.2)$$

$$\sum_{jk \in E} a_{i(jk)} x_{jk} = b_i, \forall i \in V - I \quad (5.3)$$

$$l_{ij} \leq x_{ij} \leq u_{ij}, \forall ij \in E \quad (5.4)$$

$$y_{is}^+, y_{is}^- \geq 0, \forall i \in I, \forall s \in S \quad (5.5)$$

$$x_{ij} \geq 0, \forall ij \in E$$

Označení proměnných a parametrů:

x_{ij} ... tok přes hranu $ij \in E$

l_{ij} ... dolní mez toku přes hranu $ij \in E$

u_{ij} ... horní mez toku přes hranu $ij \in E$

c_{ij} ... cena jednotky toku skrz hranu $ij \in E$

b_{is} ... požadavek vrcholu $i \in V$ pro příslušný scénář $s \in S$

$b_{is} > 0$ pro cíl

$b_i = 0$ pro přepravní uzel

$b_i < 0$ pro zdroj

y_{is} ... vyrovnávací kapacita pro vrchol $i \in V$ a scénář $s \in S$

y_{is}^+ ... při nedostatku zboží

y_{is}^- ... při nadbytku zboží

q_i ... vyrovnávací cena pro vrchol $i \in V$

q_i^+ ... při nedostatku zboží

q_i^- ... při nadbytku zboží

p_s ... pravděpodobnost daného scénáře $s \in S$

S ... množina všech možných scénářů

$a_{i(jk)}$... prvky incidenční matice grafu $i \in V, jk \in E$

E ... množina hran

V ... množina vrcholů

$I \subset V$... množina cílových vrcholů

5.7 Aplikační příklad

Navazuje se na příklad představený v kapitole 4 s komplikací v podobě náhodnosti poptávky. K řešení je použit scénářový přístup.

Možné scénáře poptávky $b(i, s)$, $i = 5, 6, 7$ s pravděpodobnostmi $p(s)$ viz Tab. 5.1.

Tab. 5.1

	$s=1$	$s=2$	$s=3$
$b(5,s)$	22	25	20
$b(6,s)$	13	10	17
$b(7,s)$	27	23	19
$p(s)$	0,5	0,3	0,2

Implementace v GAMSu

\$Title: Náhodnost poptávky

Sets

```
i   prepravní uzly a zdroje      / 1,2,3,4,8,9,10,11,12 /
ci  cíle                          / 5,6,7 /
j   hrany                        / 1-5, 1-8, 2-8, 2-9, 3-5, 3-12, 4-7, 4-12,
    5-1, 5-3, 5-8, 5-9, 5-11, 6-10, 6-12, 7-4, 7-11,
    7-12, 8-1, 8-2, 8-5, 9-2, 9-5, 9-10, 10-6, 10-9,
    11-5, 11-7, 12-3, 12-4, 12-6, 12-7 /
s   scénare / 1*3 / ;
```

Parameters

```
c(j)  distribuční nakalady přes hranu j
/ 1-5 63, 1-8 38, 2-8 33, 2-9 57, 3-5 35, 3-12 59, 4-7 90, 4-12 72,
5-1 63, 5-3 35, 5-8 45, 5-9 39, 5-11 15, 6-10 23, 6-12 77, 7-4 90, 7-11 137,
7-12 118, 8-1 38, 8-2 33, 8-5 45, 9-2 57, 9-5 39, 9-10 41, 10-6 23, 10-9 41,
11-5 15, 11-7 77, 12-3 59, 12-4 72, 12-6 77, 12-7 118 /
```

```
b(i)  kapacita uzlu i
/ 1 -17, 2 -10, 3 -13, 4 -20 /
```

```
p(s)  pravděpodobnost jednotlivých scénaru s
/ 1 0.5, 2 0.3, 3 0.2 /
```

```
qplus(ci)  kompenzační cena jednotky pro vrchol i
/ 5 500, 6 500 ,7 500/
```

```
qminus(ci)  kompenzační cena jednotky pro vrchol i
/ 5 500, 6 500 ,7 500/ ;
```

Table A(i,j) incidencní matice grafu

```
      1-5 1-8 2-8 2-9 3-5 3-12 4-7 4-12 5-1 5-3 5-8 5-9 5-11 6-10 6-12 7-4
1     -1  -1                                1
2                                     -1  -1
3                                     -1  -1
4                                     -1  -1
8           1   1
9           1
10          1
11          1
12          1   1
```

+

```

+
      7-11 7-12 8-1 8-2 8-5 9-2 9-5 9-10 10-6 10-9 11-5 11-7 12-3 12-4 12-6 12-7
1      1
2      1      1
3
4      1
8      -1 -1 -1
9      -1 -1 -1      1
10     1 -1 -1
11     1      -1 -1
12     1      -1 -1 -1 -1

```

Table AA(ci,j)

```

      1-5 1-8 2-8 2-9 3-5 3-12 4-7 4-12 5-1 5-3 5-8 5-9 5-11 6-10 6-12 7-4
5      1      1      -1 -1 -1 -1 -1
6
7      1      -1 -1
+
      7-11 7-12 8-1 8-2 8-5 9-2 9-5 9-10 10-6 10-9 11-5 11-7 12-3 12-4 12-6 12-7
5      1      1      1
6      1
7     -1 -1      1      1

```

Table bb(ci,s) kapacita uzlu ci pro scenar s

```

      1  2  3
5     22 25 20
6     13 10 17
7     27 23 19 ;

```

scalar

```

deltaP  prepinač pro variabilitu poptavky (scenare)
        /1/
deltaD  prepinač pro variabilitu počtu hran
        /0/ ;

```

Variable

```

z       celkove naklady ;

```

Positive Variable

```

x(j)   objem prepravy pres hranu j
yplus(ci,s) kompenzacni rozhodnuti
yminus(ci,s) kompenzacni rozhodnuti ;

```

Binary Variable

```

delta(j) zda dojde k vybudovani nove hrany ;

```

Equations

```

ucelfce, omez1(i), omez2(ci,s), omez3(j);

ucelfce .. z =e= sum((j), c(j)*x(j)) + deltaP*(sum((s), p(s)*(sum((ci), qplus(ci))
*yplus(ci,s) + qminus(ci)*yminus(ci,s)))) ;

omez1(i) .. sum(j, A(i,j)*x(j)) =e= b(i) ;

omez2(ci,s) .. sum(j, AA(ci,j)*x(j)) + deltaP*(yplus(ci,s) - yminus(ci,s)) =e= bb
(ci,s) ;

omez3(j) .. x(j) =l= (1-deltaD)*100 + deltaD*100*delta(j) ;

```

Model transport /all/ ;

Solve transport using mip minimizing z ;

Display x.l, z.l ;

Optimální řešení

Výstup v GAMSu:

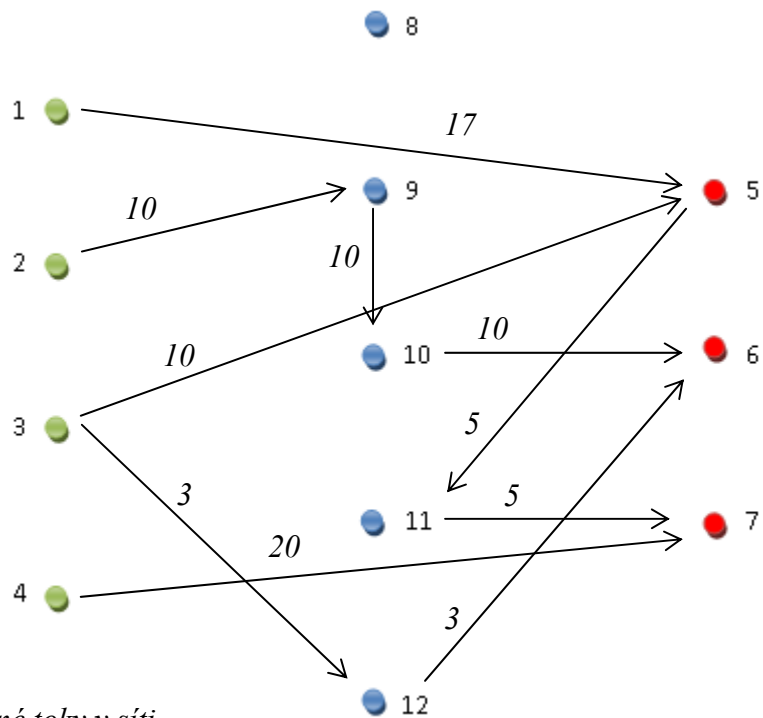
```
---- 108 VARIABLE x.L  objem prepravy pres hranu j

1-5 17.000,   2-9 10.000,   3-5 10.000,   3-12 3.000,   4-7 20.000
5-11 5.000,   9-10 10.000,  10-6 10.000,  11-7 5.000,   12-6 3.000

---- 108 VARIABLE z.L                               =      8199.000  celkove naklady
```

Pro přehlednost jsou toky znázorněny do grafu *Obr. 5.1*. Celkové minimalizované náklady jsou $z = 8199$.

Proti předchozím příkladům dopravního a přepravního modelu jsme navíc uvažovali náhodnost poptávky a její nesplnění jsme penalizovali. Tato penalizace významně zvýšila hodnotu účelové funkce. Protože v Uničově poptávka náhodně kolísá směrem nahoru od původní hodnoty uvažované poptávky (nyní hodnoty 20, 22, 25 a dříve hodnota 20) a ve Šternberku tato hodnota kolísá dolů (nyní 19, 23, 27 a dříve 27) je logické, že v optimálním řešení v Uničově bude ponechána hodnota 5 (dříve 3) a do Šternberka bude dovezeno 5 přes Brničko (dříve 7). Toto je jediná odlišnost mezi řešením původní deterministické úlohy a scénářové stochastické úlohy. S ohledem na to, že uvedené řešení je optimální pro stochastickou úlohu, použití řešení deterministické úlohy v podmínkách náhodné poptávky by ještě více zvýšilo hodnotu účelové funkce, tj. náklady.



Obr. 5.1 – Výsledné toky v síti

6 Problém přidání hran

6.1 Formulace úlohy

Mějme obecnou přepravní úlohu. Může pak nastat případ, kdy je výhodnější (levnější) postavit novou silnici (hranu), než přepravovat náklad oklikou.

Tedy do našeho modelu přidáváme náklady na vybudování nové hrany d_{ij} mezi vrcholem i a j . Proměnná $\delta_{ij} \in \{0,1\}$ nám pak značí, zda k vystavění nové hrany došlo $\delta_{ij} = 1$, či nedošlo $\delta_{ij} = 0$.

Předpokládáme konstantní náklady c_{ij} .

6.2 Matematický model

Minimalizujeme náklady na přepravu:

$$z = \sum_{ij \in E} c_{ij}x_{ij} + \sum_{ij \in E} d_{ij}\delta_{ij} \quad (6.1)$$

při splnění omezujících podmínek:

$$\sum_{jk \in E} a_{i(jk)}x_{jk} = b_i, \forall i \in V \quad (6.2)$$

$$l_{ij} \leq x_{ij} \leq u_{ij}, \forall ij \in E \quad (6.3)$$

$$l_{ij} \leq x_{ij} \leq u_{ij}\delta_{ij}, \forall ij \in E \quad (6.4)$$

$$x_{ij} \geq 0, \forall ij \in E \quad (6.5)$$

$$\delta_{ij} \in \{0,1\}, \forall ij \in E$$

Označení proměnných a parametrů:

x_{ij} ... tok přes hranu $ij \in E$

l_{ij} ... dolní mez toku přes hranu $ij \in E$

u_{ij} ... horní mez toku přes hranu $ij \in E$

c_{ij} ... cena jednotky toku skrz hranu $ij \in E$

d_{ij} ... náklad na vybudování hrany mezi vrcholy i a j , $i, j \in V$

δ_{ij} ... značí, zda k vystavení nové hrany mezi vrcholy i a j , $i, j \in V$ došlo

$\delta_{ij} = 1$, pokud k vybudování hrany došlo

$\delta_{ij} = 0$, k vybudování nedošlo

b_i ... požadavek vrcholu $i \in V$

$b_i > 0$ pro cíl

$b_i = 0$ pro přepravní uzel

$b_i < 0$ pro zdroj

$a_{i(jk)}$... prvky incidenční matice grafu $i \in V, jk \in E$

E ... množina hran

V ... množina vrcholů

6.3 Aplikační příklad

Navazuje se na příklad představený v kapitole 4 s komplikací v podobě variability počtu hran.

Přeprava mezi jednotlivými městy může probíhat i přes cesty které byly přes zimu uzavřeny, a to za malou investici, která je přivede do znovu provozuschopného stavu. Nutná investice do rekonstrukce je:

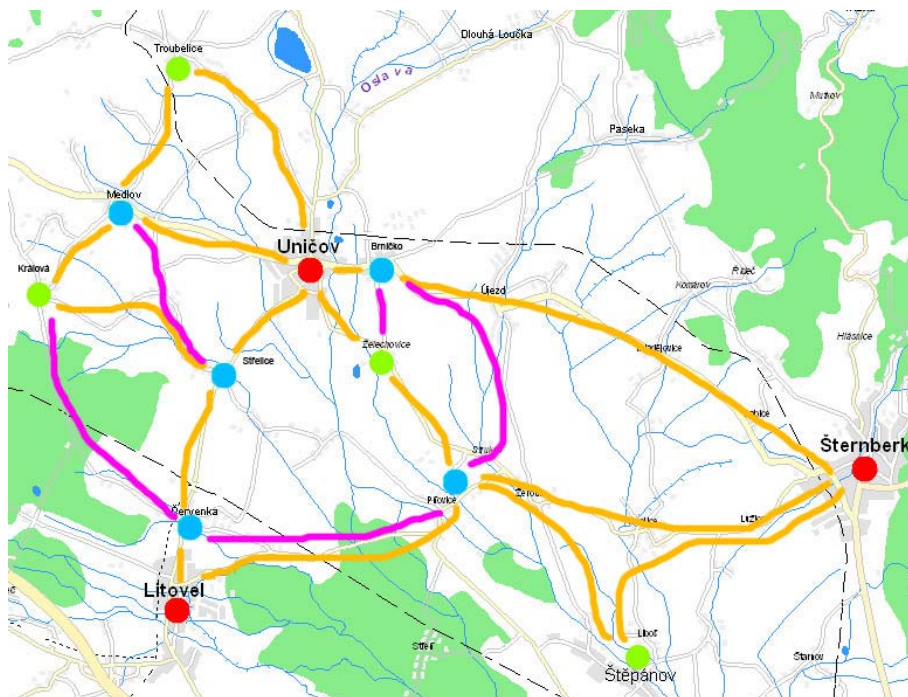
Medlov – Střelice ...	60
Králová – Červenka ...	56
Brníčko – Želechovice ...	70
Brníčko – Pňovice ...	90
Červenka – Pňovice ...	123

Hrany určené k rekonstrukci viz mapa *Obr.6.1* a jeho převedení do grafu viz *Obr.6.2*.

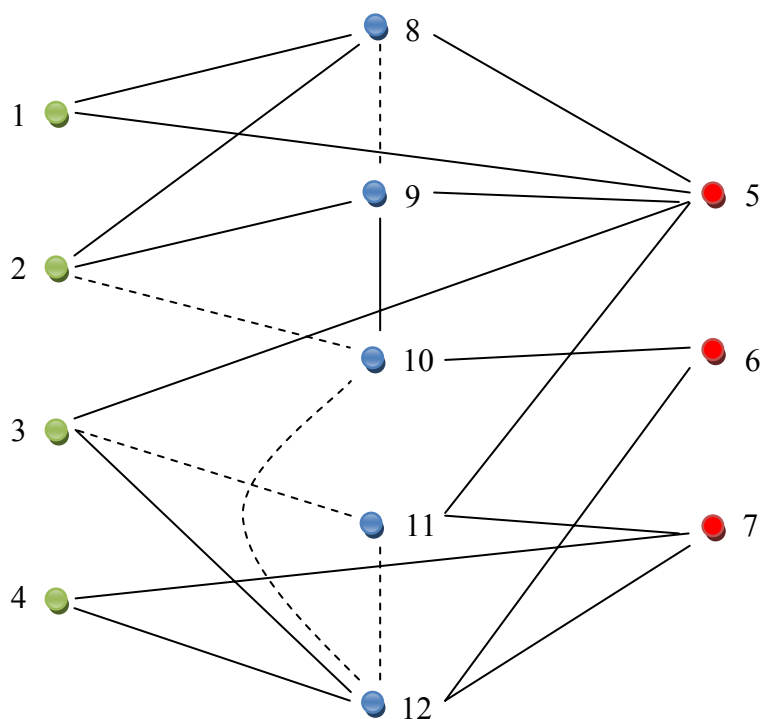
Náklady na přepravu mezi uzly viz *Tab.6.1*.

Tab. 6.1 – Přepravní náklady

	1	2	3	4	5	6	7	8	9	10	11	12
1					63			38				
2								33	57	70		
3					35						20	59
4							90					72
5	63		35					45	39		15	
6										23		77
7				90							137	118
8	38	33			45				49			
9		57			39			49		41		
10		70				23			41			66
11			20		15		77					103
12			59	72		77	118			66	103	



Obr. 6.1 – Naznačení hran, fialová jsou hrany, které můžeme znovu zprovoznit



Obr. 6.2 – Převedení mapy na graf, přičemž po hraně se můžeme pohybovat oběma směry. Možná hrana na přidání je značena čárkovaně.

Implementace v GAMSu

\$Title: Varibialita poctu hran

Sets

```
i  uzly    / 1*12 /
j  hrany   / 1-5, 1-8, 2-8, 2-9, 3-5, 3-12, 4-7, 4-12,
           5-1, 5-3, 5-8, 5-9, 5-11, 6-10, 6-12, 7-4, 7-11,
           7-12, 8-1, 8-2, 8-5, 9-2, 9-5, 9-10, 10-6, 10-9,
           11-5, 11-7, 12-3, 12-4, 12-6, 12-7,
           8-9, 2-10, 11-3, 11-12, 10-12, 9-8, 10-2, 3-11, 12-11, 12-10 / ;
```

Parameters

```
b(i)  kapacita uzlu i
      /1 17, 2 10, 3 13, 4 20, 5 20, 6 13, 7 27 /

c(j)  distribucni nakalady pres hranu j
      / 1-5 63, 1-8 38, 2-8 33, 2-9 57, 3-5 35, 3-12 59, 4-7 90, 4-12 72,
      5-1 63, 5-3 35, 5-8 45, 5-9 39, 5-11 15, 6-10 23, 6-12 77, 7-4 90, 7-11 137,
      7-12 118, 8-1 38, 8-2 33, 8-5 45, 9-2 57, 9-5 39, 9-10 41, 10-6 23, 10-9 41,
      11-5 15, 11-7 77, 12-3 59, 12-4 72, 12-6 77, 12-7 118, 8-9 49, 2-10 70,
      11-3 20, 11-12 103, 10-12 66, 9-8 49, 10-2 70, 3-11 20, 12-11 103, 12-10 66 /

d(j)  naklady na vybudovani hrany j (neni uvedeno - hrana je jiz vybudovana)
      / 8-9 60, 2-10 56, 11-3 70, 11-12 90, 10-12 123
      9-8 60, 10-2 56, 3-11 70, 12-11 90, 12-10 123 /;
```

Table A(i,j) incidencni matice grafu

```
1-5 1-8 2-8 2-9 3-5 3-12 4-7 4-12 5-1 5-3 5-8 5-9 5-11 6-10 6-12 7-4
1  -1 -1                                     1
2                                     -1 -1
3                                     -1 -1                                     1
4                                     -1 -1                                     1
5  1                                     1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
6                                     -1 -1
7                                     1                                     -1
8  1 1                                     1
9                                     1
10                                     1
11                                     1
12                                     1 1                                     1

+
7-11 7-12 8-1 8-2 8-5 9-2 9-5 9-10 10-6 10-9 11-5 11-7 12-3 12-4 12-6 12-7
1                                     1
2                                     1 1
3                                     1
4                                     1
5                                     1 1
6                                     1
7 -1 -1                                     1
8                                     -1 -1 -1
9                                     -1 -1 -1 1
10                                    1 -1 -1
11  1                                     -1 -1
12                                     1 -1 -1 -1

+
8-9 2-10 11-3 11-12 10-12 9-8 10-2 3-11 12-11 12-10
1
2 -1                                     1
3 1                                     -1
4
5
6
7
8 -1                                     1
```

```

    9    1          -1          -1          1
    10         1          -1          -1          1          1
    11         -1         -1          1          1          1
    12          1          1          1          -1         -1 ;

scalar
    deltaP  prepinac pro variabilitu poptavky
            /0/
    deltaD  prepinac pro variabilitu poctu hran
            /1/ ;

Variable
    z       celkove naklady ;

Positive Variable
    x(j)    objem prepravy pres hranu j ;

Binary Variable
    delta(j) zda dojde k vybudovani nove hrany ;

Equations
    ucelfce, omez1(i), omez2(j);

    ucelfce .. z =e= sum((j), c(j)*x(j)) + deltaD*(sum((j), d(j)*delta(j))) ;

    omez1(i) .. sum(j, A(i,j)*x(j)) =e= b(i) ;

    omez2(j) .. x(j) =l= (1-deltaD)*100 + deltaD*100*delta(j) ;

Model transport /all/ ;

Solve transport using mip minimizing z ;

Display x.l, z.l ;

```

Optimální řešení

Výstup v GAMSu:

```

----      100 VARIABLE x.L  objem prepravy pres hranu j

1-5  14.000,    1-8   3.000,    4-7  20.000,    9-10  3.000,    10-6  13.000
11-5  6.000,    11-7  7.000,    8-9   3.000,    2-10  10.000,    3-11  13.000

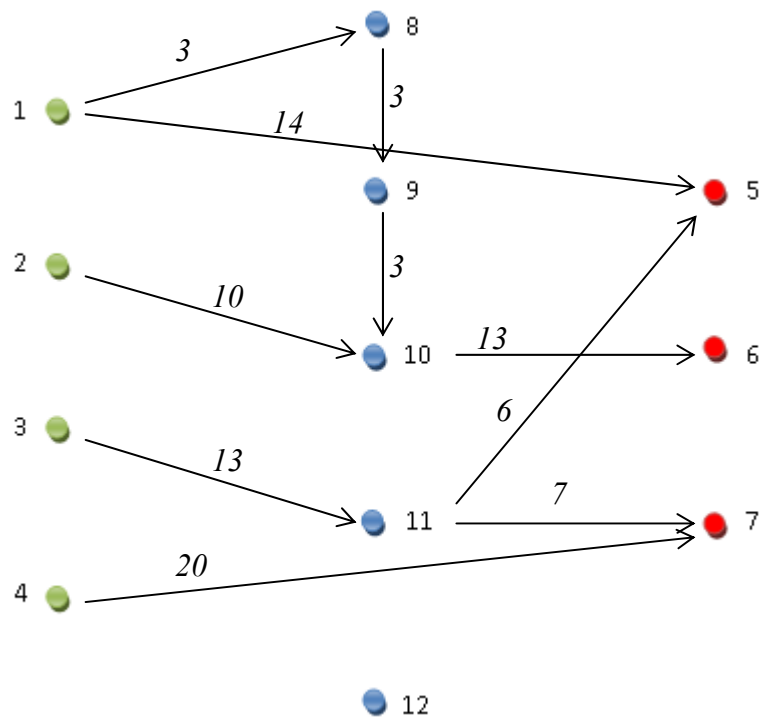
----      100 VARIABLE z.L              =      5140.000  celkove naklady

```

Pro přehlednost jsou toky znázorněny do grafu *Obr. 6.3*. Celkové minimalizované náklady jsou $z = 5140$.

S ohledem na nižší nastavení nákladů na otevření cest po zimní přestávce, byly využity pro dopravu tři další hrany. To umožnilo snížit celkové přepravní náklady a došlo k zásadním změnám v přepravních plánech. Například dosud dopravené množství z Troubelic do Uničova pokračovalo částečně přes Brničko na Šternberk. S ohledem na otevřenou cestu z Želechovic do Brnička je tato cesta využita k dopravě jak do Šternberka, tak do Uničova (je třeba upozornit, že náklady na cestu do Uničova přes Brničko nebo přímo jsou s ohledem na kvalitu komunikací stejné, není vhodné se nechat zmást pouze mapou s umístěním cest). Rovněž zásobování

Litovle je s ohledem na možnost využití otevření cest z Medlova a Králové řešeno efektivnějším způsobem. Z hlediska počtu hran nejslibnější přepravní uzel Pňovice již není využit.



Obr. 6.3 – Výsledné toky v síti

7 Variabilita poptávky i počtu hran

7.1 Formulace úlohy

Vycházíme z obecné přepravní úlohy a přidáváme problémy řešené v kapitole 5 a 6.

Tedy v modelu uvažujeme náhodnou poptávku a existuje možnost postavit nové cesty (hrany) mezi jednotlivými uzly.

Do modelu obecné přepravní úlohy zapracujeme kompenzační rozhodnutí y_{is} příslušnému danému vrcholu $i \in I$, kde I je množina cílů, a scénáři $s \in S$ s příslušnou pravděpodobností p_s a náklady na vybudování hrany d_{ij} mezi vrcholem i a j , $i, j \in V$.

7.2 Matematický model

Minimalizujeme

$$z = \sum_{ij \in E} c_{ij} x_{ij} + \sum_{ij \in E} d_{ij} \delta_{ij} + \sum_{s \in S} p_s \left(\sum_{i \in I} q_i^+ y_{is}^+ + \sum_{i \in I} q_i^- y_{is}^- \right) \quad (7.1)$$

Za podmínek

$$\sum_{jk \in E} a_{i(jk)} x_{jk} + y_{is}^+ - y_{is}^- = b_{is}, \forall s \in S, \forall i \in I \quad (7.2)$$

$$\sum_{jk \in E} a_{i(jk)} x_{jk} = b_i, \forall s \in S, \forall i \in V - I \quad (7.3)$$

$$l_{ij} \leq x_{ij} \leq u_{ij} \delta_{ij}, \forall ij \in E \quad (7.4)$$

$$x_{ij} \geq 0, \forall ij \in E \quad (7.5)$$

$$y_{is}^+, y_{is}^- \geq 0, \forall i \in I, \forall s \in S \quad (7.6)$$

$$\delta_{ij} \in \{0, 1\}, \forall ij \in E \quad (7.7)$$

Označení proměnných a parametrů:

x_{ij} ... tok přes hranu $ij \in E$

l_{ij} ... dolní mez toku přes hranu $ij \in E$

u_{ij} ... horní mez toku přes hranu $ij \in E$

c_{ij} ... cena jednotky toku skrz hranu $ij \in E$

d_{ij} ... náklad na vybudování hrany mezi vrcholy i a j , $i, j \in V$

δ_{ij} ... značí, zda k vystavení nové hrany mezi vrcholy i a j , $i, j \in V$ došlo

$\delta_{ij} = 1$ pokud k vybudování hrany došlo

$\delta_{ij} = 0$ k vybudování nedošlo

b_{is} ... požadavek vrcholu $i \in V$ pro příslušný scénář $s \in S$

$b_{is} > 0$ pro cíl

$b_i = 0$ pro přepravní uzel

$b_i < 0$ pro zdroj

y_{is} ... vyrovnávací kapacita pro vrchol $i \in V$ a scénář $s \in S$

y_{is}^+ ... při nedostatku zboží

y_{is}^- ... při nadbytku zboží

q_i ... vyrovnávací cena pro vrchol $i \in V$

q_i^+ ... při nedostatku zboží

q_i^- ... při nadbytku zboží

p_s ... pravděpodobnost daného scénáře $s \in S$

S ... množina všech možných scénářů

$a_{i(jk)}$... prvky incidenční matice grafu $i \in V$, $jk \in E$

E ... množina hran

V ... množina vrcholů

$I \subset V$... množina cílových vrcholů

7.3 Zobecněná formulace úlohy

Vycházíme z modelu 7.2 a přidáme tzv. přepínače Δ_d a Δ_p . Jejich pomocí, tzn. nastavením jejich hodnot na nulu nebo jedničku, budeme moci přepínat mezi modely, které byly popsány výše.

Tedy mezi modely:

- obecným přepravním problémem

$$(\Delta_d = 0, \Delta_p = 0)$$

- obecným přepravním problémem s náhodností poptávky

$$(\Delta_d = 0, \Delta_p = 1)$$

- obecným přepravním problémem s možností přidávání hran

$$(\Delta_d = 1, \Delta_p = 0)$$

- obecným přepravním problémem s náhodností poptávky i možností přidávání hran

$$(\Delta_d = 1, \Delta_p = 1)$$

7.4 Zobecněný matematický model

Minimalizujeme

$$z = \sum_{ij \in E} c_{ij} x_{ij} + \Delta_d \sum_{ij \in E} d_{ij} \delta_{ij} + \Delta_p \sum_{s \in S} p_s \left(\sum_{i \in V} q_i^+ y_{is}^+ + \sum_{i \in V} q_i^- y_{is}^- \right) \quad (7.8)$$

Za podmínek

$$\sum_{jk \in E} a_{i(jk)} x_{jk} + y_{is}^+ - y_{is}^- = b_{is}, \forall s \in S, \forall i \in I \quad (7.9)$$

$$\sum_{jk \in E} a_{i(jk)} x_{jk} = b_i, \forall s \in S, \forall i \in V - I \quad (7.10)$$

$$l_{ij} \leq x_{ij} \leq (1 - \Delta_d) u_{ij} + \Delta_d u_{ij} \delta_{ij}, \forall ij \in E \quad (7.11)$$

$$x_{ij} \geq 0, \forall ij \in E \quad (7.12)$$

$$y_{is}^+, y_{is}^- \geq 0, \forall i \in I, \forall s \in S \quad (7.13)$$

$$\delta_{ij} \in \{0,1\}, \forall ij \in E \quad (7.14)$$

7.5 Aplikační příklad

Navazuje se na příklad představený v kapitole 4 s komplikacemi variability počtu hran a náhodnosti poptávky současně. Data použijeme stejná jako v kapitolách 5 a 6.

Implementace v GAMSu

\$Title: Náhodnost poptavky i variabilita počtu hran

Sets

```
i  prepravni uzly a zdroje    / 1,2,3,4,8,9,10,11,12 /
ci cile                       / 5,6,7 /
j  hrany                      / 1-5, 1-8, 2-8, 2-9, 3-5, 3-12, 4-7, 4-12,
                             5-1, 5-3, 5-8, 5-9, 5-11, 6-10, 6-12, 7-4, 7-11,
                             7-12, 8-1, 8-2, 8-5, 9-2, 9-5, 9-10, 10-6, 10-9,
                             11-5, 11-7, 12-3, 12-4, 12-6, 12-7
                             8-9, 2-10, 11-3, 11-12, 10-12, 9-8, 10-2, 3-11, 12-11, 12-10 /
s  scenare / 1*3 / ;
```

Parameters

```
c(j)  distribucni nakalady pres hranu j
/ 1-5 63, 1-8 38, 2-8 33, 2-9 57, 3-5 35, 3-12 59, 4-7 90, 4-12 72,
5-1 63, 5-3 35, 5-8 45, 5-9 39, 5-11 15, 6-10 23, 6-12 77, 7-4 90, 7-11 137,
7-12 118, 8-1 38, 8-2 33, 8-5 45, 9-2 57, 9-5 39, 9-10 41, 10-6 23, 10-9 41,
11-5 15, 11-7 77, 12-3 59, 12-4 72, 12-6 77, 12-7 118, 8-9 49, 2-10 70,
11-3 20, 11-12 103, 10-12 66, 9-8 49, 10-2 70, 3-11 20, 12-11 103, 12-10 66 /

b(i)  kapacita uzlu i
/ 1 -17, 2 -10, 3 -13, 4 -20 /

p(s)  pravdepodobnost jednotlivych scenaru s
/ 1 0.5, 2 0.3, 3 0.2 /

qplus(ci)  kompenzacni cena jednotky pro vrchol i
/ 5 500, 6 500 ,7 500/

qminus(ci) kompenzacni cena jednotky pro vrchol i
/ 5 500, 6 500 ,7 500/

d(j)  naklady na vybudovani hrany j (neni uvedeno - hrana je jiz vybudovana)
/ 8-9 60, 2-10 56, 11-3 70, 11-12 90, 10-12 123
9-8 60, 10-2 56, 3-11 70, 12-11 90, 12-10 123 /;
```

Table A(i,j) incidenční matice grafu

```
1-5 1-8 2-8 2-9 3-5 3-12 4-7 4-12 5-1 5-3 5-8 5-9 5-11 6-10 6-12 7-4
1  -1  -1                                     1
2                                     -1  -1
3                                     -1  -1                                     1
4                                     -1  -1                                     1
8          1  1                                     1
9                                     1
10                                     1
11                                     1
12          1  1                                     1

+
7-11 7-12 8-1 8-2 8-5 9-2 9-5 9-10 10-6 10-9 11-5 11-7 12-3 12-4 12-6 12-7
1                                     1
2                                     1  1
3                                     1
4                                     1
8                                     -1  -1  -1
9                                     -1  -1  -1  1
10          1  -1  -1  1  -1  -1
11          1  1                                     -1  -1
12          1  1                                     -1  -1  -1  -1

+
8-9 2-10 11-3 11-12 10-12 9-8 10-2 3-11 12-11 12-10
1
2          -1                                     1
3          1                                     -1
```

```

4
8 -1
9 1
10 1 -1 -1 -1 -1 1
11 -1 -1 1 1
12 1 1 -1 -1

```

Table AA(ci,j) incidencni matice grafu

```

1-5 1-8 2-8 2-9 3-5 3-12 4-7 4-12 5-1 5-3 5-8 5-9 5-11 6-10 6-12 7-4
5 1
6
7
+
7-11 7-12 8-1 8-2 8-5 9-2 9-5 9-10 10-6 10-9 11-5 11-7 12-3 12-4 12-6 12-7
5 1 1
6 1 1
7 -1 -1 1 1
+
8-9 2-10 11-3 11-12 10-12 9-8 10-2 3-11 12-11 12-10
5
6
7

```

Table bb(ci,s) kapacita uzlu ci pro scenar s

```

1 2 3
5 22 25 20
6 13 10 17
7 27 23 19 ;

```

scalar

```

deltaP prepinac pro variabilitu poptavky (scenare)
/1/
deltaD prepinac pro variabilitu poctu hran
/1/ ;

```

Variable

```

z celkove naklady ;

```

Positive Variable

```

x(j) objem prepravy pres hranu j
yplus(ci,s) kompenzacni rozhodnuti
yminus(ci,s) kompenzacni rozhodnuti ;

```

Binary Variable

```

delta(j) zda dojde k vybudovani nove hrany ;

```

Equations

```

ucelfce, omez1(i), omez2(ci,s), omez3(j);

ucelfce .. z =e= sum((j), c(j)*x(j)) + deltaP*(sum((s), p(s)*(sum((ci),
qplus(ci)*yplus(ci,s) + qminus(ci)*yminus(ci,s))))
+ deltaD*(sum((j), d(j)*delta(j))) ;

omez1(i) .. sum(j, A(i,j)*x(j)) =e= b(i) ;

omez2(ci,s) .. sum(j, AA(ci,j)*x(j)) + deltaP*(yplus(ci,s) - yminus(ci,s))
=e= bb(ci,s) ;

omez3(j) .. x(j) =l= (1-deltaD)*100 + deltaD*100*delta(j) ;

```

Model transport /all/ ;

Solve transport using mip minimizing z ;

Display x.l, z.l ;

Optimální řešení

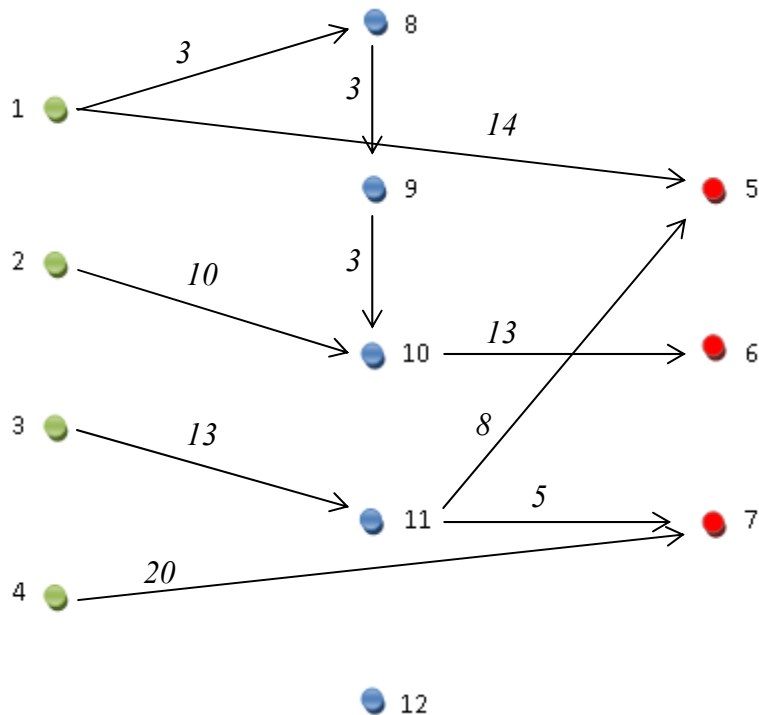
Výstup v GAMSu:

```
---- 131 VARIABLE x.L   objem prepravy pres hranu j

1-5 14.000,   1-8  3.000,   4-7 20.000,   9-10 3.000,   10-6 13.000
11-5 8.000,   11-7 5.000,   8-9  3.000,   2-10 10.000,   3-11 13.000

---- 131 VARIABLE z.L           = 7916.000 celkove naklady
```

Pro přehlednost jsou toky znázorněny do grafu *Obr. 7.1*. Celkové minimalizované náklady jsou $z = 7916$. Protože jsme umožnili využití dalších hran za nízké náklady, je hodnota účelové funkce nižší než hodnota účelové funkce předchozí úlohy stochastického programování. Protože jsme umožnili využívání v zimě zavřených cest, struktura optimálního řešení, tj. použité cesty jsou stejné jako v kapitole 6. Protože uvažujeme náhodnost, mění se ovšem přepravované množství do Uničova z Brníčka (zvyšuje se z 6 na 8 kvůli náhodnému růstu poptávky) a naopak se snižuje přepravované množství z Brníčka do Šternberka (kvůli náhodnému poklesu poptávky) ze 7 na 5.



Obr. 7.1 – Výsledné toky v síti

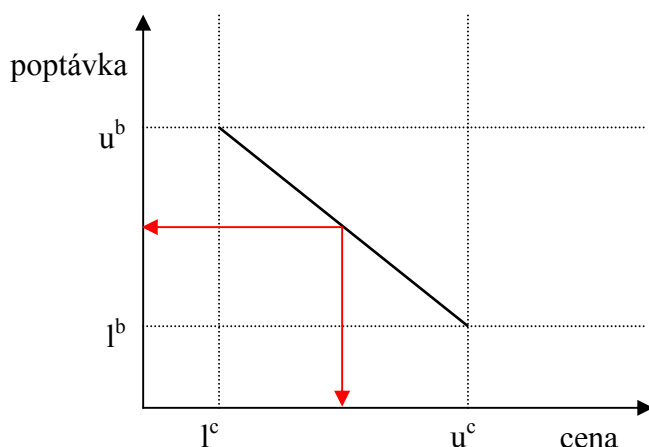
8 Dynamické oceňování

V této kapitole jsem se inspirovala přednáškami prof. Haugena z norské univerzity v Molde, kterých jsem se zúčastnila při jeho návštěvě FSI VUT Brno v létě roku 2009. Více lze nalézt o dané problematice v textech [12], [13].

Probírá v nich problém tzv. „dynamic pricing“, což překládám jako problém dynamického oceňování. Prof. Haugen tento přístup aplikuje na jiné optimalizační problémy, já jsem se rozhodla pro původní aplikaci tohoto přístupu na náš problém. Můžeme použít lineární aproximaci tak jako prof. Haugen.

8.1 Lineární model

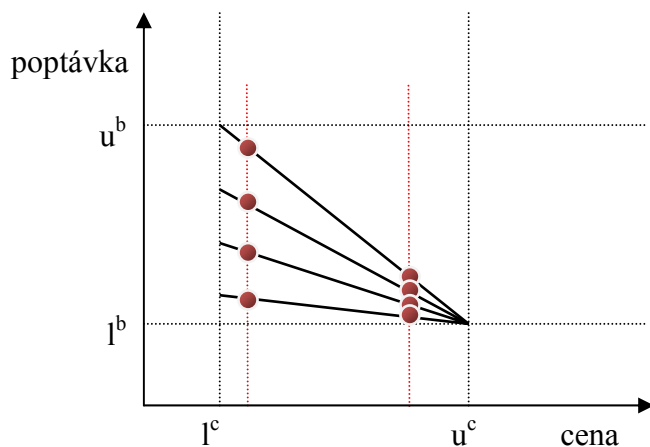
Změna poptávky v závislosti na ceně má spíše hyperbolický průběh, ale pro úzký vyšetřovaný interval použijeme lineární aproximaci. Viz *Obr.8.1*.



Obr. 8.1 – Lineární průběh závislosti poptávky na ceně

Růst ceny c_{ij} omezujeme horní hranicí l_{ij}^c a její pokles dolní hranicí u_{ij}^c , $\forall ij \in E$. Pak samozřejmě nastává i omezení poptávky hranicemi $l_{is}^b \leq b_{is} \leq u_{is}^b$, $\forall i \in V$ a $\forall s \in S$.

Budeme-li uvažovat scénářový případ poptávky, předpokládáme, na základě konzultací s prof. Haugenem, že s poklesem ceny narůstá rozptyl poptávky. Viz *Obr. 8.2*.



Obr. 8.2 – Příklad více scénářů průběhu poptávky

8.2 Formulace úlohy

Máme danou poptávku $b_{is}(c_i^*)$ jako lineární funkci ceny c_i^* , kde c_i^* bereme jako průměrnou cenu jednotky toku $\overline{c_{j \rightarrow i}}$ přicházející do vrcholu $i \in I$ z vrcholu $j \in V$.

Z lineárního modelu se poptávka rovná $b_{is} = \alpha_{is} - \beta_{is}c_i^*$, kde β_{is} je směrnice přímky závislosti poptávky b_{is} na ceně c_i^* , $\forall i \in I, \forall s \in S$.

8.3 Matematický model

Minimalizujeme

$$z = \sum_{ij \in E} c_{ij}x_{ij} + \Delta_d \sum_{ij \in E} d_{ij}\delta_{ij} + \Delta_p \sum_{s \in S} p_s \left(\sum_{i \in I} q_i^+ y_{is}^+ + \sum_{i \in I} q_i^- y_{is}^- \right) \quad (8.1)$$

Za podmínek

$$\sum_{jk \in E} a_{i(jk)}x_{jk} + \Delta_p(y_{is}^+ - y_{is}^-) = \alpha_{is} - \beta_{is}c_i^*, \forall s \in S, \forall i \in I \quad (8.2)$$

$$\sum_{jk \in E} a_{i(jk)}x_{jk} = b_i, \forall s \in S, \forall i \in V - I \quad (8.3)$$

$$l_{ij} \leq x_{ij} \leq (1 - \Delta_d)u_{ij} + \Delta_d u_{ij} \delta_{ij}, \forall ij \in E \quad (8.4)$$

$$x_{ij} \geq 0, \forall ij \in E \quad (8.5)$$

$$y_{is}^+, y_{is}^- \geq 0, \forall i \in I, \forall s \in S \quad (8.6)$$

(8.7)

$$0 < l_{ij}^c \leq c_{ij} \leq u_{ij}^c, \forall ij \in E$$

(8.8)

$$c_i^* = \overline{c_{j \rightarrow i}}, \forall i \in I, \forall j \in V$$

(8.9)

$$\delta_{ij} \in \{0,1\}, \forall ij \in E$$

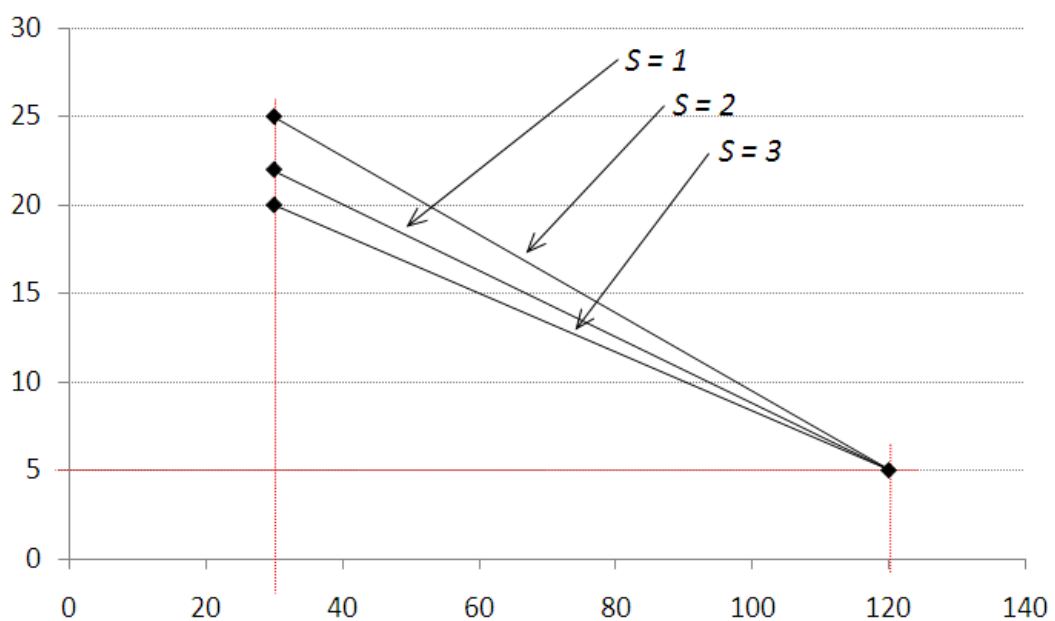
8.4 Aplikační příklad

Navazuje se na příklad představený v kapitole 4 s komplikacemi variability počtu hran a náhodnosti poptávky současně s přidáním problému dynamického oceňování.

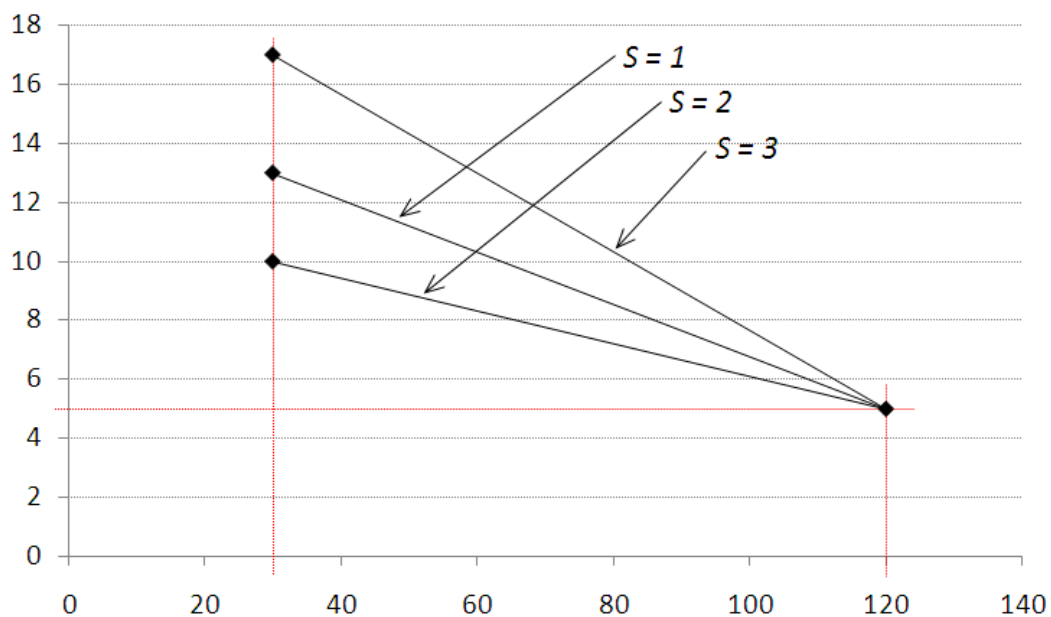
Města Uničov, Litovel a Šternberk se podílejí na programu snižování nezaměstnanosti, která je na Olomoucku tradičně vysoká a dostávají dotace od státu, pokud denně zaměstnají minimálně 5 lidí bez práce.

Náhodná cena za přepravu jednoho člověka se pohybuje v intervalu $\langle 30, 120 \rangle$.

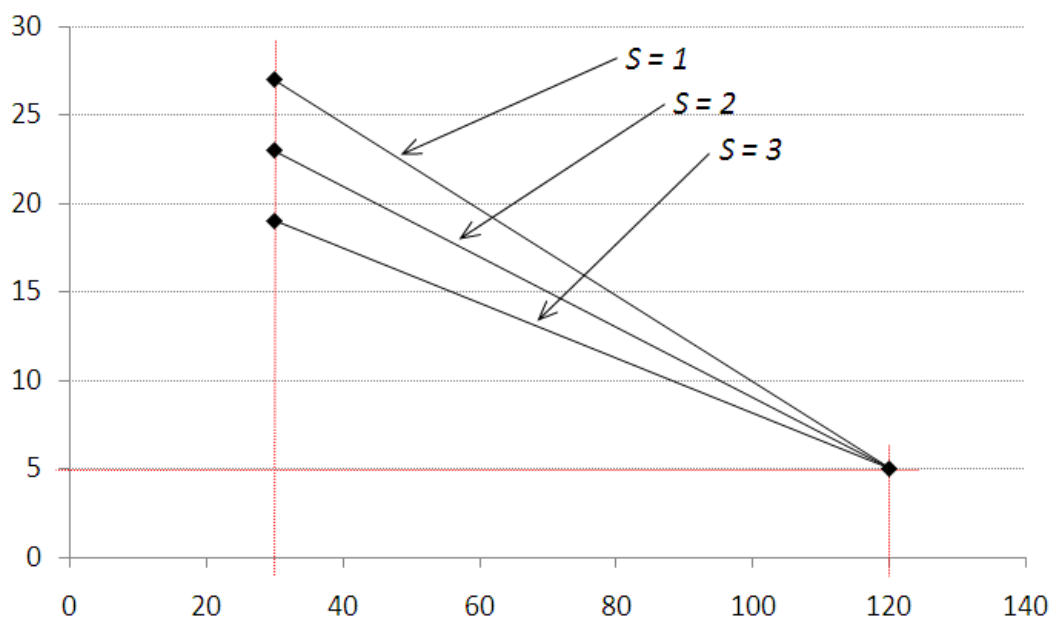
Průběh závislosti poptávky na ceně pro vrchol 5 viz *Obr.8.3*, pro vrchol 6 viz *Obr.8.4*, pro vrchol 7 viz *Obr.8.5*



Obr. 8.3 – Průběh poptávky pro vrchol 5 a scénář s



Obr. 8.4 – Průběh poptávky pro vrchol 6 a scénář s



Obr. 8.5 – Průběh poptávky pro vrchol 7 a scénář s

Vyčíslení směrníc α_{is} , β_{is} viz Tab.8.1.

Tab.8.1

	i=5			i=6			i=7		
	s=1	s=2	s=3	s=1	s=2	s=3	s=1	s=2	s=3
α_{is}	27,67	35	25	15,67	11,67	21	34,33	29	23,67
β_{is}	0,19	0,33	0,17	0,09	0,06	0,13	0,24	0,2	0,16

Implementace v GAMSu

\$Title: Dynamic pricing

Sets

```
i  prepravni uzly a zdroje / 1,2,3,4,8,9,10,11,12 /
ci cile / 5,6,7 /
j  hrany / 1-5, 1-8, 2-8, 2-9, 3-5, 3-12, 4-7, 4-12,
      5-1, 5-3, 5-8, 5-9, 5-11, 6-10, 6-12, 7-4, 7-11,
      7-12, 8-1, 8-2, 8-5, 9-2, 9-5, 9-10, 10-6, 10-9,
      11-5, 11-7, 12-3, 12-4, 12-6, 12-7
      8-9, 2-10, 11-3, 11-12, 10-12, 9-8, 10-2, 3-11, 12-11, 12-10 /
s  scenare / 1*3 / ;
```

Parameters

```
b(i) kapacita uzlu i
/ 1 -17, 2 -10, 3 -13, 4 -20 /

p(s) pravdepodobnost jednotlivych scenaru s
/ 1 0.5, 2 0.3, 3 0.2 /

qplus(ci) kompenzacni cena jednotky pro vrchol i
/ 5 500, 6 500 ,7 500/

qminus(ci) kompenzacni cena jednotky pro vrchol i
/ 5 500, 6 500 ,7 500/

d(j) naklady na vybudovani hrany j (neni uvedeno - hrana je jiz vybudovana)
/ 8-9 60, 2-10 56, 11-3 70, 11-12 90, 10-12 123
  9-8 60, 10-2 56, 3-11 70, 12-11 90, 12-10 123 /;
```

Table A(i,j) incidencni matice grafu

```
      1-5 1-8 2-8 2-9 3-5 3-12 4-7 4-12 5-1 5-3 5-8 5-9 5-11 6-10 6-12 7-4
1     -1  -1
2
3
4
8      1   1
9
10
11
12
+
      7-11 7-12 8-1 8-2 8-5 9-2 9-5 9-10 10-6 10-9 11-5 11-7 12-3 12-4 12-6 12-7
1
2
3
4
8
9
10
11
12
+
      8-9 2-10 11-3 11-12 10-12 9-8 10-2 3-11 12-11 12-10
1
2
3
4
8
9
10
11
12
```

Table AA(ci,j) *incidenční matice grafu*

	1-5	1-8	2-8	2-9	3-5	3-12	4-7	4-12	5-1	5-3	5-8	5-9	5-11	6-10	6-12	7-4
5	1				1				-1	-1	-1	-1	-1			
6														-1	-1	
7						1										-1

+

	7-11	7-12	8-1	8-2	8-5	9-2	9-5	9-10	10-6	10-9	11-5	11-7	12-3	12-4	12-6	12-7
5					1		1				1					
6									1							1
7	-1	-1										1				1

+

	8-9	2-10	11-3	11-12	10-12	9-8	10-2	3-11	12-11	12-10
5										
6										
7										

Table bb(ci,s) *kapacita uzlu ci pro scenar s*

	1	2	3
5	22	25	20
6	13	10	17
7	27	23	19

Table alfa(ci,s) *vysec*

	1	2	3
5	27.67	35	25
6	15.67	11.67	21
7	34.33	29	23.67

Table beta(ci,s) *smernice*

	1	2	3
5	0.19	0.33	0.17
6	0.09	0.06	0.13
7	0.24	0.2	0.16

 ;

scalar

deltaP *prepinac pro variabilitu poptavky (scenare)*
 /1/
 deltaD *prepinac pro variabilitu poctu hran*
 /1/ ;

Variable

z *celkove naklady* ;

Positive Variable

x(j) *objem prepravy pres hranu j*
 yplus(ci,s) *kompensacni rozhodnuti*
 yminus(ci,s) *kompensacni rozhodnuti*
 c(j) *náhodná cena jednotky toku pres hranu j*
 cprum(ci) *prumerna cena toku prichazejici do vrcholu ci* ;

Binary Variable

delta(j) *zda dojde k vybudovani nove hrany* ;

Equations

ucelfce,omez1(i),omez2(ci,s),omez3(j),omez4(j),omez5(j),prum5,prum6,prum7;
 ucelfce .. z =e= **sum**((j), c(j)*x(j)) + deltaP*(**sum**((s), p(s)*(**sum**((ci),
 qplus(ci)*yplus(ci,s) + qminus(ci)*yminus(ci,s))))
 + deltaD*(**sum**((j), d(j)*delta(j))) ;
 omez1(i) .. **sum**(j, A(i,j)*x(j)) -e- b(i) ;
 omez2(ci,s) .. **sum**(j, AA(ci,j)*x(j)) + deltaP*(yplus(ci,s) - yminus(ci,s))

```

=e=  alfa(ci,s)-beta(ci,s)*cprum(ci) ;

omez3(j) .. x(j) =l= (1-deltaD)*100 + deltaD*100*delta(j) ;

omez4(j) .. 30 =l= c(j) ;

omez5(j) .. c(j) =l= 120 ;

prum5 .. cprum('5') =e= (c('1-5')+c('3-5')+c('8-5')+c('9-5')+c('11-5'))/5 ;

prum6 .. cprum('6') =e= (c('10-6')+c('12-6'))/2 ;

prum7 .. cprum('7') =e= (c('11-7')+c('4-7')+c('12-7'))/3 ;

Model transport / all / ;

Solve transport using minlp minimizing z ;

Display x.l, z.l ;

```

Optimální řešení

Výstup v GAMSu:

```

----      149 VARIABLE x.L   objem prepravy pres hranu j

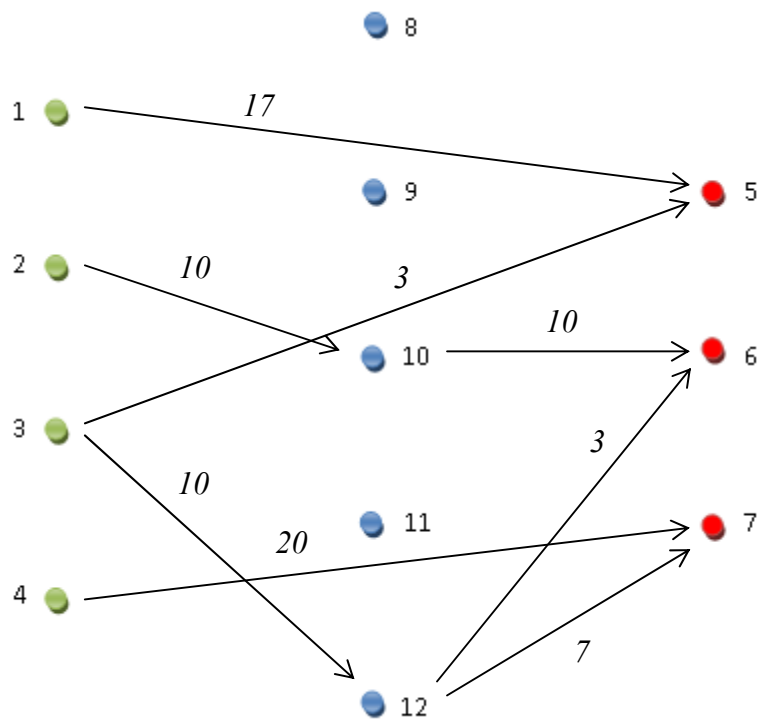
1-5  17.000,    3-5   2.900,    3-12 10.100,    4-7  20.000,    10-6 10.000
12-6  2.970,    12-7  7.130,    2-10 10.000

----      149 VARIABLE z.L                               =      5208.421   celkove naklady

```

Jak vidíme, došlo k neceločíselnosti výsledků. To ošetříme zaokrouhlováním.

Pro přehlednost jsou toky znázorněny do grafu *Obr. 8.6*. Celkové minimalizované náklady jsou $z = 5208,421$. S ohledem na nastavení hodnot parametrů optimální řešení úlohy využilo možnosti snižovat nebo zvyšovat ceny přepravy v určitém intervalu k tomu, aby významně snížilo předešlou optimální hodnotu průměrných nákladů. Bylo upřednostněno snížení cen i za cenu nárůstu variability poptávky v uzlech. Z nabízených cest k otevření byla využita pouze jedna, která zlevňuje dopravu do Litovle. Díky snížení nákladů na dopravu byla uspokojena vyšší poptávka ve Šternberku a zároveň nižší poptávka v Uničově. Model zahrnující náhodnost poptávky, využívání dalších cest a možnost měněním ceny přepravy ovlivňovat poptávku představuje výpočtově náročnou úlohu celočíselného nelineárního programování se scénáři. Použití tohoto modelu poskytlo řešení, které má přehlednější strukturu, než v předchozích stochastických úlohách.



Obr. 8.6 - Výsledné toky v síti

9 Závěr

Ve své práci jsem se zaměřila na formulaci přepravní úlohy, její zobecňování a modifikace. Jednotlivé modely jsem implementovala v programu GAMS. Na model dopravní úlohy jsem navázala modelem speciální přepravní úlohy. Ten jsem zobecnila pro případ jednoduchého grafu. Modely jsem implementovala v systému GAMS pomocí reprezentace grafu incidenční maticí. To mi umožnilo na uvedený model navazovat dalšími zobecněními bez komplikovaných změn.

Nejprve jsem v obecné přepravní úloze uvažovala náhodnost poptávky, kterou jsem modelovala pomocí scénářů. Zvolila jsem HN přístup k náhodnosti v modelu. Řešení získané na aplikačním příkladě mi umožnilo srovnání s předcházejícím deterministickým případem. Dále jsem se zabývala možností využití dalších hran k přepravě. Vhodným modelem se ukázala úloha celočíselného lineárního programování. Spojením předchozích dvou modelů vznikl model kombinující náhodnost poptávky a využití dalších hran. S ohledem na zvolený postup je uvedený model opět úlohou celočíselného lineárního programování. Snížení hodnot účelové funkce s ohledem na možnost využití dalších cest klade ovšem nároky na realizovaný výpočet tím, že místo úlohy lineárního programování odpovídající scénářové stochastické úloze musí být řešena náročnější úloha celočíselného lineárního programování.

Původním modelem v diplomové práci je model, který navíc zahrnuje dynamické oceňování. Tento model pro rozsáhlejší úlohy představuje náročnější výpočetní nároky, protože se jedná o úlohu celočíselného nelineárního programování se scénáři. Pro modelování závislosti poptávky na ceně jsem volila lineární model podle prof. Haugena. Proto jsem volila aplikační problém odpovídající velikostí (řešené úlohy zahrnovaly stovky omezení a reálných proměnných, a dále desítky proměnných celočíselných podle volby dat). Zvolená velikost dat mi dále umožnila vizualizovat část vstupních dat a výsledků přehledně pomocí grafů a map. Všechny aplikační úlohy byly implementovány a vyřešeny pomocí systému GAMS. Vytvořené zdrojové kódy lze snadno upravit pro jiné datové soubory.

Při řešení diplomové práce byly využity poznatky projektů výzkumného centra CQR MŠMT ČR č. 1M06047 a dále subprojektů B/CZ0046/2/0021 a B/CZ0046/3/0035 projektu EHS NVF ČR a zejména přístup dynamického oceňování bude využit při modelování optimalizačních úloh v dalších inženýrských aplikacích, kde je nutné zohlednit závislost náhodného parametru na rozhodnutí, viz projekty GAČR č.103/08/1658 a výzkumný záměr MŠMT ČR č.MSM0021630519

Seznam použitých zdrojů

- [1] Dopravní problém [online] [cit 2010-03-12]. Dostupný z <<http://www.mti.tul.cz/files/oa/linprog/doprav.htm>>
- [2] Dopravní problém [online] [cit 2010-03-12]. Dostupný z <<http://www1.osu.cz/studium/mopv2/doprav/#Reseni>>
- [3] Dopravní problém [online] [cit 2010-03-12]. Dostupný z <http://www.vspj.cz/~kubisova/studenti/MEK/soubory/13_prednaska.doc>
- [4] Hliněný, Petr. Teorie grafů. 2008. 125s. [online] [cit 2009-12-12]. Dostupný z <<http://www.fi.muni.cz/~hlineny/Vyuka/GT/Grafy-text07.pdf>>
- [5] Dopravní problém [online] [cit 2010-03-12]. Dostupný z <http://cs.wikipedia.org/wiki/Dopravn%C3%AD_prob%C3%A9m>
- [6] Dopravní problém [online] [cit 2010-03-12]. Dostupný z <http://water.fce.vutbr.cz/vyuka/cP03/dopravni_problem.pdf>
- [7] Fryšová, D., Hantych, P., Charamza, P., Kadlčák, R., Klicnar, M., Mydlo, M., Pavlicová, M., Popela, P., Tlustý, P., Večeře, J.: *Modelovací systém GAMS*. MFF UK, 1993. 128 s.
- [8] Rosenthal, Richard E.: *GAMS – A User's Guide*. GAMS Development Corporationm Washington, DC, USA, 2008. 281 s.
- [9] Šeda, Miloš. Teorie grafů. 2003. 89 s. [online] [cit 2009-12-12]. Dostupný z <http://www.uai.fme.vutbr.cz/~mseda/TG03_MS.pdf>
- [10] Popela, P.: *Stochastic programming*. UoM, 2004
- [11] Logistika. [online] [cit 2010-04-25]. Dostupné z <<http://cs.wikipedia.org/wiki/Logistika>>
- [12] Kjetil K. Haugen, Asmund Olstad, Bård I. Pettersen: *Solving Large-scale Profit Maximization Capacitated Lot-size Problems by Heuristic Methods*.
- [13] Kjetil K. Haugen, Asmund Olstad, Bård I. Pettersen: *The profit maximizing capacitated lot-size (PCLSP) problem*.
- [14] PJsoft InfoMapa 14
- [15] Nash, S.G, Sofer, A.: *Linear and Nonlinear Programming*. International Editions, 1996. 692 s.
- [16] Laporte, F.: *Introduction to Logistics Systems Planning and Control*. JohnWiley & Sons Ltd, 2004. 352 s.
- [17] Klapka, J., Dvořák, J., Popela, P.: *Metody operačního výzkumu*. VUTIUM, 2001. 165 s.

Seznam použitých symbolů

$\overline{c_{j \rightarrow i}}$	průměrná hodnota ceny toků ze všech $j \in V$ do $i \in I$
Δ_d	přepínač pro variabilitu hran
Δ_p	přepínač pro náhodnost poptávky
$a_{i(jk)}$	prvek matice incidence
b_i	kapacita vrcholu i
b_{is}	kapacita vrcholu i příslušná scénáři s
c_{ij}	cena jednotky toku skrz hranu ij
d_{ij}	náklady na vybudování hrany ij
l_i^b	dolní hranice poptávky vrcholu i
l_{ij}	dolní hranice toku přes hranu ij
l_{ij}^c	dolní hranice ceny hrany ij
l_{is}^b	dolní hranice poptávky vrcholu i a scénáře s
p_s	pravděpodobnost příslušná scénáři s
q_i^+	vyrovnávací cena pro vrchol i při nedostatku zboží
q_i^-	vyrovnávací cena pro vrchol i při nadbytku zboží
u_i^b	horní hranice poptávky vrcholu i
u_{ij}	horní hranice toku přes hranu ij
u_{ij}^c	horní hranice ceny hrany ij
u_{is}^b	horní hranice poptávky vrcholu i a scénáře s
x_{ij}	tok přes hranu ij
y_{is}^+	vyrovnávací kapacita pro vrchol i a scénář s při nedostatku zboží
y_{is}^-	vyrovnávací kapacita pro vrchol i a scénář s při nadbytku zboží
$A^I(G)$	matice incidence příslušná grafu G
$A^S(G)$	matice sousednosti příslušná grafu G
N_0	přirozená čísla s 0
\mathbf{R}^n	n -rozměrný prostor reálných čísel
\mathbf{b}^T	vektor kapacit
\mathbf{c}^T	vektor
δ_{ij}	značí, zda dojde k vybudování hrany ij
C	cílové uzly
G	graf
P	přepravní uzly
S	množina scénářů
Z	zdrojové uzly
i	označení vrcholů
ij	hrana mezi vrcholy i a j
j	označení vrcholů
s	scénář
z	účelová funkce
A	matice incidence
E	množina hran

I	množina cílových vrcholů
N	přirozená čísla
R	reálná číslo
V	množina vrcholů
x	vektor toků
ξ	vektor náhodných elementů