ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FAKULTA DOPRAVNÍ

Kozlovska Olha

TYPOLOGIE DISKRÉTNÍCH LOKAČNÍCH ÚLOH

Bakalářská práce

2019

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta dopravní
děkan
Konviktská 20, 110 00 Praha 1

K617.. Ústav logistiky a managementu dopravy

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení studenta (včetně titulů):

Olha Kozlovska

Kód studijního programu a studijní obor studenta:

B 3710 – LOG – Logistika a řízení dopravních procesů

Název tématu (česky): **Typologie diskrétních lokačních úloh**

Název tématu (anglicky): Discrete Facility Location Problems Typology

Zásady pro vypracování

Při zpracování bakalářské práce se řiďte osnovou uvedenou v následujících bodech:

- lokační úloha - její definice a význam pro logistické plánování,
- matematický model lokační úlohy - popis jednotlivých částí modelu, význam jednotlivých skupin strukturálních podmínek,
- rešerše odborné časopisecké literatury zabývající se lokačními úlohami,
- softwarové nástroje k řešení lokačních úloh,
- praktické ukázky řešení vybraných typů lokačních úloh s využitím softwarových nástrojů.

Rozsah grafických prací: podle pokynů vedoucího bakalářské práce

Rozsah průvodní zprávy: minimálně 35 stran textu (včetně obrázků, grafů a tabulek, které jsou součástí průvodní zprávy)

Seznam odborné literatury: Farahani, R., Z.; Hekmatfar, M.: Facility Location: Concepts, Models, Algorithms and Case Studies. Physica Verlag, New York, 2009

Eiselt, H., A.; Sandblom, C., L.: Decision Analysis, Location Models, and Scheduling Problems. Springer Verlag, Berlin, 2004

Vedoucí bakalářské práce: **doc. Ing. Dušan Teichmann, Ph.D.**

Datum zadání bakalářské práce: **30. června 2018**

(datum prvního zadání této práce, které musí být nejpozději 10 měsíců před datem prvního předpokládaného odevzdání této práce vyplývajícího ze standardní doby studia)

Datum odevzdání bakalářské práce: **26. srpna 2019**

a) datum prvního předpokládaného odevzdání práce vyplývající ze standardní doby studia a z doporučeného časového plánu studia
b) v případě odkladu odevzdání práce následující datum odevzdání práce vyplývající z doporučeného časového plánu studia

L. S.

doc. Ing. Tomáš Horák, Ph.D.
vedoucí
Ústavu logistiky a managementu dopravy

doc. Ing. Pavel Hrubeš, Ph.D.
děkan fakulty

Potvrzuji převzetí zadání bakalářské práce.

Olha Kozlovska
jméno a podpis studenta

V Praze dne .... 30. června 2018

Prohlášení

Nemám závažný důvod proti užívání tohoto školního díla ve smyslu § 60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

V Praze dne : 26.08.2019

podpis

Poděkování

Ráda bych poděkovala všem těm, díky kterým vznikla daná práce. Především svému vedoucímu doc. Ing. Dušanu Teichmannu, Ph.D. za vedení bakalářské práce a připomínky k řešení úloh a k celkovému zpracování.

Nakonec bych chtěla ještě poděkovat svým rodičům a celé rodině za morální a materiální podporu po celou dobu mého studia na této univerzitě.

Abstrakt

Bakalářská práce představuje přehled 18 reprezentativních úloh lokační analýzy. V teoretické části jsou definovány různé typy diskrétních lokačních úloh a jejich matematické zápisy s rozšířeným vystvětlením a srovnáním. Byly přezkoumány standardní úlohy, jako jsou kapacitní a nekapacitní model s lokalizací zdrojů, úlohy o umístění p-center a p-mediánů, a méně známé úlohy, které se objevily v posledních letech. Praktickou část nahrazuje řešení lokačních úloh v optimalizačním softwaru a zobrazení výsledku s využitím náhodných hodnot.

Klíčová slova

Diskretní lokační úloha, optimalizační software Xpress-IVE, matematický model, dopravní síť, optimalizace.

Abstract

This bachelor thesis presents a survey of 18 representative problems in location research. In the theoretical part, different types of discrete facility location problems and their mathematical notation with extended explanation and comparison are defined. Standard problems such as simple uncapacitated/capacitated facility location, center, median location problems, and less know problems that appeared in recent years were reviewed. The practical part is replaced by solving location problems in optimization software and representing the result using random values.

Keywords

Discrete facility location problem, optimization software Xpress-IVE, mathematical model, transport network, optimization.

OBSAH

1. Úvod

Rozmístění zdrojů v lokalitách je kritickým aspektem strategického plánování pro široké spektrum veřejných a soukromých firem. Zřízení a provoz každého zdroje je velmi drahá a dlouhodobá záležitost, což vede k tomu, že rozhodnutí o počtu, velikosti a kapacitě zdrojů může výrazně ovlivnit nejenom budoucí příjmy z komerční činnosti, ale i obecně konkurenceschopnost firem. Plánování optimálního rozmístění patří mezi nejvýznamnější rozhodovací problémy každé distribuční firmy, ať už se jedná o maloobchodní řetězec, který uvažuje o rozmístění nových prodejen, nebo o velkou společnosti, která uvažuje o rozmístění regionálních skladů.

Lokační úloha byla poprvé formálně vytvořena Alfredem Weberem v roce 1909, který zkoumal problém lokalizace jednoho skladu tak, aby hodnota souhrné vzdálenosti mezi tímto skladem a skupinou zákazníků byla minimální. Po tomto počátečním výzkumu byla problematika lokačních úloh rozvíjena různými směry, které nenavazovaly na ucelenou teori. Více teoretického zájmu o lokalizační problémy vyvolala seminární práce Hakimiho v roce 1964, který se zabýval rozmístěním ústředen v komunikační síti a policejních stanic v dálničním systému. Jeho publikace zkoumala lokalizaci jednoho nebo více zdrojů v síti tak, aby se minimalizovala celková vzdálenost mezi zákazníky a jejich nejbližším zdrojem nebo aby se minimalizovala maximální vzdálenost mezi zákazníkem a zdrojem. Od té doby začal probíhat systematický výzkum v dané oblasti.

Lokační úlohu je možno charakterizovat následujícími hlavními prvky: prostor, v němž mají být zdroje rozmístěny, zdroje existující a nové, zákazníci, kteří vyjadřují poptávku po službě včetně její intenzity, interakce mezi zákazníky i zdroji, metrika pro měření vzdálenosti mezi zákazníky a zdroji a omezení, která mají být splněna.

Vzhledem možnému umístění v zájmovém území lze lokační úlohy rozdělit na tři základní druhy. Pokud existuje konečný počet lokalit vhodných k umístění zdrojů v zájmovém území, tak se jedná o tzv. diskrétní lokační úlohu. Pokud mohou být zdroje rozmístěny v libovolném místě celého zájmového území, tak se jedná o tzv. spojitou lokační úlohu. A pokud lze zdroje rozmísťovat na struktuře odpovídající dopravní síti, potom se jedná o tzv. síťovou lokační úlohu. Bakalářská práce je zaměřena na diskrétní lokační úlohy a jejich řešení s využitím matematického programování.

V dané bakalářské práci bude představen reprezentativní přehled diskrétních lokačních úloh. Nejdřív bude definována a zkoumána nejjednodušší lokační úloha. Dále bude prezentován optimalizační software Xpress-IVE určený k řešení úloh matematického programování, tedy i řešení lokačních úloh. Následující kapitoly budou věnovány typologii a podrobnějšímu popisu dalších typů lokačních úloh, které byly dohledány v odborné literatuře. Vzhledem k tomu, že při tvorbě bylo čerpáno pouze z cizojazyčných zdrojů, budou jednotlivé typy lokačních úloh nazývány pouze jejich anglickými ekvivalenty.

2. Lokační úloha

2.1 Obecný popis lokační úlohy

Je dána dopravní síť. V zadané síti je je známa množina lokalit I vhodných pro umístění zdroje a množina zákazníků J. Každý zákazník Z_j, kde $j \in J$ a, má definován svůj určitý požadavek b_j za řešené období a každá lokalita L_i, kde $i \in I$, ve které můžeme umístit zdroj (sklad, výrobní závod, depo, hub, překladiště nebo distribuční centrum) má známu svou kapacitu a_i (v kapacitně neomezených úlohách nejsou hodnoty a_i definovány). Zprovoznění libovolného zdroje v lokalitě L_i vyvolává fixní náklady f_i. Pokud existuje relace mezi Z_j a L_i, tak vznikají náklady c_{ij} reprezentující náklady na přepravu 1 jednotky zboží z lokality L_i k zákazníkovi Z_j.

Cílem je zvolit nejvhodnější lokality pro umístění zdrojů a přiřadit zákazníky k těmto zdrojům tak, aby se minimalizovala hodnota celkových nákladů na provoz umístěných zdrojů a na zásobování zákazníků.

Za účelem řešení úlohy budou zavedeny dvě skupiny bivalentních proměnných označených jako y_i a x_{ij}. Nechť y_i je bivalentní proměnná reprezentující provozování zdroje v lokalitě L_i, kde $i \in I$. Pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě L_i, kde $i \in I$, v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě L_i, kde $i \in I$, v provozu není. Nechť x_{ij} je bivalentní proměnná reprezentující přiřazení zákazníka Z_j, kde $j \in J$, ke zdroji provozovanému v lokalitě L_i, kde $i \in I$. Pokud je po ukončení optimalizačního výpočtu $x_{ij} = 1$, potom je zákazník Z_j, kde $j \in J$, přiřazen ke zdroji provozovaném v lokalitě L_i, kde $i \in I$, pokud je $x_{ij} = 0$, potom zákazník Z_j, kde $j \in J$, není přiřazen ke zdroji provozovanému v lokalitě L_i, kde $i \in I$.

Nejčastěji se model kapaictně neomezené lokační úlohy zapisuje ve tvaru:

$$min\, f(x,y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \qquad (1)$$

za podmínek:

$$\sum_{i \in I} x_{ij} = 1 \qquad \text{pro } j \in J \qquad (2)$$

$$x_{ij} \leq y_i \qquad \text{pro } i \in I, j \in J \qquad (3)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (4)$$

$$x_{ij} \in \{0,1\} \qquad \text{pro } i \in I, j \in J \qquad (5)$$

Funkce (1) je účelovou funkcí modelu a je složená ze dvou částí. První část zahrnuje náklady spojené s přiřazením zákazníků ke zdrojům v lokalitách. Druhá část reprezentuje náklady spojené s otevřením zdrojů v lokalitách. Skupina podmínek (2) je množinou klíčových podmínek modelu, které zapezpečují, že každý zákazník bude přiřazen. Kdyby tyto podmínky nebyly zařazeny v modelu, tak hodnota účelové funkce v modelu bude nulová, protože žádný zákazník by nebyl přiřazen žádnému zdroji (což souvisí s požadavkem na mimimalizaci nákladů v účelové funkci). Počet podmínek (2) se rovná počtu zákazníků. Skupina podmínek (3) obsahuje vazební podmínky, které dávají do vzájemného vztahu dvě bivalentní proměnné. Pokud zákazník Z_j, kde $j \in J$, je přiřazen ke zdroji v lokalitě L_i, kde $i \in I$ (matematicky vyjádřeno $x_{ij} = 1$), tak podmínka zajistí, že zdroj bude v lokalitě L_i, kde $i \in I$, v provozu (matematicky vyjádřeno $y_i = 1$) a fixní náklady na tento zdroj se započítají do účelové funkce. Pokud zdroj v lokalitě L_i, kde $i \in I$, není v provozu (matematicky vyjádřeno $y_i = 0$), tak žádný zákazník Z_j, kde $j \in J$, nebude k tomuto zdroji přidělen (matematicky vyjádřeno $x_{ij} = 0$). Bez zařazení této skupiny podmínek do modelu by se do účelové funkce nepřičítaly fixní náklady na zprovoznění zdroje a hodnota účelové funkce by byla nižší, než by byla ve skutečnosti. Počet podmínek (3) se rovná součinu počtu lokalit a zákazníků. Skupiny podmínek (4) a (5) vymezují definiční obory proměnných – jedná se o tzv. obligátní (obligatorní) podmínky. Počet podmínek (4) se rovná počtu lokalit. Počet podmínek (5) se rovná součinu počtů lokalit a zákazníků.

V případě kapacitních lokačních úloh, tedy úloh, ve kterých zdroje mají svou maximální kapacitu, se bude v soustavě omezujících podmínek vyskytovat ještě jedna množina podmínek a to:

$$\sum_{j \in J} b_j x_{ij} \leq a_i \qquad \text{pro } i \in I \qquad (6)$$

Levá část této podmínky sčítá požadavky zákazníků, které bude uspokojovat daný zdroj, pokud mu bude zákazník Z_j, kde $j \in J$, přidělen, pravá strana reprezentuje kapacitu daného zdroje. Podmínka jako celek tedy zajišťuje, že nedojde k překročení kapacity zdroj. Bez těchto podmínek by mohla nastat situace, kdy dojde k překročení kapacity zdroje. Počet podmínek (6) se rovná počtu lokalit.

Kapacitní podmínka nemusí být částí modelu, je-li zřejmé, že kapacity všech zdrojů jsou dostačující. To s určitostí nastává v situaci, kdy jsou kapacity všech zdrojů minimálně rovny součtu požadavků všech zákazníků. Když model obsahuje kapacitní podmínky, hovoří se o kapacitně omezené lokační úloze.

Celkový počet omezujících podmínek v kapacitním modelu modelu je $2|I| + |J| + 2|I||J|$.

Skupina podmínek (3) se dá zapsat ještě v ekvivalentním tvaru:

$$\sum_{j \in J} x_{ij} \leq n y_i \qquad \text{pro } i \in I \qquad (7)$$

Konstanta n reprezentuje počet zákazníků. Je také možno ji nahradit libovolnou prohibitivní konstantou, což je velmi velké číslo (větší než n). Levá strana podmínky reprezentuje celkový počet zákazníků přidělených ke zdroji v lokalitě L_i, kde $i \in I$. Kdyby na pravé straně podmínky nefigurovala konstanta n mohl by být každému zdroji přidělen maximálně jeden zákazník, což nemusí být výhodné jak z hlediska volné kapacity zdroje a souvisejících nákladů, tak dokonce i z hlediska přípustnosti úlohy.

Výhodou skupiny podmínek (3) je jejich jednoduchost při výpočtu, ale nevýhodou je velký počet podmínek, který se rovná součinu počtu lokalit a zákazníků. Skupina podmínek (7) má opačný charakter – jsou výpočetně náročnější, ale je jich menší počet rovný počtu lokalit.

V případě kapacitně omezené lokační úlohy je možno podmínky (3) a (6) nebo (6) a (7) nahradit skupinou podmínek:

$$\sum_{j \in J} b_j x_{ij} \leq a_i y_i \qquad \text{pro } i \in I \qquad (8)$$

2.2 Optimalizační software Xpress-IVE

Xpress-IVE je optimalizační software, určený pro řešení úloh lineárního programování (LP), smíšeného celočíselného lineárního programování (MILP), konvexního kvadratického programování (QP) a dalších. Programování v prostředí Xpress-IVE se uskutečnuje využitím modelovacího jazyka Mosel.

Text programu určený k řešení úloh matematického programování v prostředí Xpress-IVE můžeme pro zjednodušení rozdělit na dvě části. První částí je deklarační část, kde je zapotřebí definovat všechny proměnné, konstanty a indexy, se kterými potom bude v textu programu pracováno. První část je omezena klíčovými slovy „*declarations*“ a „*end-declarations*“. Například v úloze, ve které jsou k dispozici tři lokality pro umístění zdrojů, je možno zdroje zadefinovat následovně:

zdroj=1..3

Dále je potřeba dodržet několik pravidel pro deklarování veličin. Proměnná se označuje v software jako „*mpvar*“. V případě konstanty je to označení „*real*“. Pokud deklarujeme veličinu typu pole (vektory, matice), tak se k tomu používá označení „*array(...)of* “, kde místo teček bude uveden rozsah pole – jeden nebo více indexů vztahujících se k dané veličině a za „*of*“ se uvede druh veličiny hodnot (konstanta, proměnná), které dané pole reprezentuje. Pro ukázku zadeklarujeme dvě veličiny x_{ij} a f_i, kde x_{ij} je bivalentní proměnná, reprezentující přiřazení zákazníka $j \in J$ k zdroji v lokalitě $i \in I$, a f_i jsou fixní náklady na zprovoznění zdroje v lokalitě $i \in I$. Opět uvažujme, že v úloze se vyskytují 3 zdroje a dále 2 zákazníci.

zdroj=1..3
zakaznik=1..2
f:array(zdroj)of real
x:array(zdroj,zakaznik)of mpvar

Po deklaraci veličin nasleduje druhá část modelu a to jeho tělo. Nejdřív se zadávají hodnoty konstant, které byly předem označeny jako „*real*“. Dané hodnoty budou odlišné pro jednotlivé situace a případy. Pro demonstraci funkčnosti daných modelů budou používány náhodně modelové hodnoty, které nemají žádný vztah k realným společnostem a případům.

Dále se v textu programu uvádějí všechny omezující podmínky vyskytující se v řešeném modelu. Abychom mohli ve stručnější formě zapisovat celé skupiny podmínek, využívá se příkazu cyklu „*forall(i in index1, j in index2, ...)*“, kde *index1*, *index2* jsou předem definované rozsahy polí. Dalším důležitým příkazem je „*sum(i in index1, j in index2,...)*“, který provede součet hodnot přes daná pole. Výraz „*is_binary*“ znamená, že daná proměnná je bivalentní (může nabývat hodnot 0 nebo 1). Příkaz „*writeln(“...“)*“ se používá pro vypsání znaků na pravou část obrazovky, na které se zobrazují výsledky.

Pro zapis účelové funkce se používá libovolné označení, v bakalářské práci bude nejčastěji používáno jednotné označení účelové funkce „*celkove_naklady:=...*“, jiné označení bude využíváno ojediněle. Pokud bude úloha minimalizační, tak se za účelovou funkci zařadí příkaz „*minimize(celkove_naklady)*“, v případě maximalizační potom příkaz „*maximize(ucelova_funkce)*“. K požadavku na výpis hodnoty účelové funkce se používá příkaz „*getobjval*“, pro vypis hodnot proměnných nebo jiných výrazů obsahujících hodnoty proměnných (kromě účelové funkce) se používá příkaz „*getsol*“.

Hranice textu programu jsou vymezeny klíčovými slovy „*model*“ (pro začátek textu programu) a „*end-model*“ (pro konec textu programu).

Modelu lokační úlohy (1) – (5) popsanému v podkapitole 1.2, ve kterém jsou uvažovány 3 lokality a 3 zákazníci, odpovídá následující text programu v programovacím jazyce Mosel:

```
model Simple
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
c:array(zdroj,zakaznik)of real
f:array(zdroj)of real
x:array(zdroj,zakaznik)of mpvar
y:array(zdroj)of mpvar
end-declarations
forall(j in zakaznik)sum(i in zdroj)x(i,j)=1
forall(i in zdroj,j in zakaznik)x(i,j)<=y(i)
forall(i in zdroj,j in zakaznik)x(i,j)is_binary
forall(i in zdroj)y(i)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik)c(i,j)*x(i,j)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou : ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

```
c::  [80,95,73,                      f::  [150,165,138]
      69,82,96,
      85,76,92]
```

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 391 Kc

x(3,1)=1 *x(3,2)=1* *x(3,3)=1* *y(3)=1*

3. Modely vybraných typů lokačních úloh s jedním typem přepravované komodity

3.1 Jednoúrovňové jednokomoditní lokační úlohy

Základními úlohami optimalního rozmístění v prostoru jsou úlohy lokačního typu, ve kterých je nutno plánovat rozmístění obslužných středisek na jedné úrovni a optimalizovat přepravu homogenní komodity, tak aby celkové náklady byly minimální.

Tyto typy úloh jsou v odborné literatuře nazývány např. Single-sink Fixed Charge Transportation Problem, Aggregate Plant Transportation Problem, Capacitated Facility Location, Generalized Assignment problem, Single-source Capacitated Facility Location Problem. Z uvedené množiny úloh bude v předložené bakalářské práci prezentováno pět úloh označovaných v literatuře jako Simple Uncapacitated Facility Location, Capacitated Facility Location, Capacitated Facility Location Problem, Single-source Capacitated Facility Location Problem, Generalized Capacited Facility Location.

3.1.1 Simple Uncapacitated Facility Location (SUFL)

SUFL je jedna z nejjednoduších lokačních úloh. V SUFL je definována dopravní siť, která obsahuje množinu zákazníků a množinu lokalit vhodných pro rozmístění zdrojů, které mají neomezenou kapacitu. Všechny náklady vyskytující se v modelu jsou lineární a jsou předem známy. SUFL je velice podobný základnímu modelu lokační úlohy popsanému v podkapitole 1.2. Jediný rozdíl mezi oběma modely je ve významu proměnné x_{ij}. V základním modelu je proměnná x_{ij} bivalentní proměnnou reprezentující přidělení zákazníka zdroji v lokalitě, přičemž přidělený zdroj pak uspokojí celkový požadavek přiděleného zákazníka. V SUFL proměnná x_{ij} může nabývat hodnoty z intervalu $\langle 0; 1\rangle$, což znamená, že celkový požadavek zákazníka může být splněn z více zdrojů současně.

Zdroj: AIKENS, C [3].

Symboly v modelu mají následující významy:

I – množina lokalit vhodných pro umístění zdrojů

J – množina zákazníků

c_{ij} – celkové náklady na uspokojení celého požadavku zákazníka $j \in J$ ze zdroje v lokalitě $i \in I$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

x_{ij} – podíl požadavku zákazníka $j \in J$ uspokojeného zdrojem umístěným v lokalitě $i \in I$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj lokalitě $\mathrm{i} \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není.

Matematický model úlohy lze zapsat ve tvaru:

$$min\, f(x, y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \qquad (9)$$

za podmínek:

$$\sum_{i \in I} x_{ij} = 1 \qquad \text{pro } j \in J \qquad (10)$$

$$x_{ij} \leq y_i \qquad \text{pro } i \in I, j \in J \qquad (11)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (12)$$

$$x_{ij} \geq 0 \qquad \text{pro } i \in I\,, j \in J \qquad (13)$$

Komentář k modelu (9) – (13) je v zásadě shodný s komentářem k modelu (1) – (5). Je však uvést, že skupina omezujících podmínek (10) zajistí, že celkový požadavek všech zákazníků bude splněn (vzhledem k definičnímu oboru však neplatí, že uvedený požadavek bude splněn pouze z jednoho zdroje). Zároveň skupina podmínek (10) zajistí, že proměnné x_{ij} budou nabývat hodnot maximálně 1.

Celkový počet omezujících podmínek v modelu je 2$|I|$ + $|J|$ + 2 $|I||\,|J|$.

V literatuře se můžete ještě setkat s dalšími názvy modelu SUFL: Simple Plant-Location Problem (P. Hansen, J. Brimberg, D. Uroševic, N. Mladenovic, 2007).

Text programu pro typ lokační úlohy SUFL v programovacím jazyce Mosel má následující tvar:

```
model SUFL
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
c:array(zdroj,zakaznik)of real
f:array(zdroj)of real
x:array(zdroj,zakaznik)of mpvar
y:array(zdroj)of mpvar
end-declarations
forall(j in zakaznik)sum(i in zdroj)x(i,j)=1
forall(i in zdroj,j in zakaznik)x(i,j)<=y(i)
forall(i in zdroj,j in zakaznik)x(i,j)>=0
forall(i in zdroj)y(i)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik)c(i,j)*x(i,j)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

c::	*[80,95,73, 69,82,96, 85,76,92]*	*f::*	*[150,165,138]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 391 Kc

x(3,1)=1 *x(3,2)=1* *x(3,3)=1* *y(3)=1*

3.1.2 Capacitated Facility Location (CFL)

V CFL je definována dopravní síť, která obsahuje množinu zákazníků a množinu lokalit vhodných pro umístění zdrojů, což je obdobou modelu SUFL s tím rozdílem, že daná úloha uvažuje omezené kapacity zdrojů. To znamená, že každý zdroj v lokalitě má definováno maximální množství komodity. Cílem úlohy je zvolit nejvhodnější rozmístění zdrojů v lokalitách k tomu určených a přiřadit zákazníky zprovozněným zdrojům tak, aby hodnota celkových nákladů na provoz zprovozněných zdrojů a na zásobování zákazníků a za podmínky nepřekročení kapacity zprovozněných zdrojů byla minimální.

Zdroj: AARDAL, K., Y. POCHET a L. A. WOLSEY [1]

Symboly v modelu mají následující významy:

I – množina lokalit vhodných pro umístění zdrojů

J – množina zákazníků

c_{ij} – celkové přepravní náklady mezi zdrojem v lokalitě $i \in I$ a zákazníkem $j \in J$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

a_i – kapacita zdroje v lokalitě $i \in I$

b_j – požadavek zákazníka $j \in J$

x_{ij} – tok komodity mezi zdrojem zprovozněným v lokalitě $i \in I$ a zákazníkem $j \in J$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není.

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x, y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \tag{14}$$

za podmínek

$$\sum_{i \in I} x_{ij} = b_j \qquad \text{pro } j \in J \qquad (15)$$

$$\sum_{j \in J} x_{ij} \le a_i y_i \qquad \text{pro } i \in I \qquad (16)$$

$$x_{ij} \ge 0 \qquad \text{pro } i \in I, j \in J \qquad (17)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (18)$$

Funkce (14) je účelovou funkcí modelu a je složená ze dvou částí. První část reprezentuje přepravní náklady mezi zákazníky a zdroji. Druhá část jsou náklady spojené se zprovozněním zdrojů v lokalitách. Skupina podmínek (15) zabezpečuje, že celkový požadavek zákazníka bude splněn. Počet podmínek (15) se rovná počtu zákazníků. Skupina podmínek (16) zajistí, že kapacity zprovozněných zdrojů nebudou překročeny. Počet podmínek (16) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupiny podmínek (17) a (18) vymezují definiční obory proměnných použitých v modelu. Počet podmínek (17) se rovná součinu počtu počtů lokalit vhodných pro umístění zdrojů a zákazníků, počet podmínek (18) se rovná počtu lokalit vhodných pro umístění zdrojů.

Celkový počet omezujících podmínek v modelu je 2 $|I|$+ $|J|$ + 2$|I|$ $|J|$.

V literatuře se můžete ještě setkat s dalšími názvy modelu CFL: Capacitated Warehouse Location Problem (C. R.Mateus a C. T.Bornstein(1991)).

Text programu pro typ lokační úlohy CFL v programovacím jazyce Mosel má následující tvar:

```
model CFL
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
c:array(zdroj,zakaznik)of real
x:array(zdroj,zakaznik)of mpvar
f:array(zdroj)of real
y:array(zdroj)of mpvar
b:array(zakaznik)of real
a:array(zdroj)of real
end-declarations
forall(j in zakaznik)sum(i in zdroj)x(i,j)=b(j)
forall(i in zdroj)sum(j in zakaznik)x(i,j)<=a(i)*y(i)
forall(i in zdroj)y(i)is_binary
forall(i in zdroj, j in zakaznik)x(i,j)>=0
celkove_naklady:=sum(i in zdroj,j in zakaznik)c(i,j)*x(i,j)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

```
c:: [80,95,73,          f:: [150,165,138]
     69,82,96,          b:: [200,195,215]
     85,76,92]          a:: [410,390,450]
```

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 44768 Kc

x(1,3)=215 x(2,1)=200 x(3,2)=195 y(1)=1 y(2)=1 y(3)=1

3.1.3 Capacited Facility Location Problem (CFLP)

Úloha CFLP je obdobou úlohy CFL. Rozdíl mezi nimi je v podstatě pouze formální, protože zatímco v modelu úlohy CFL proměnné x_{ij} reprezentují absolutní toky zboží mezi zdroji a zákazníky, v modelu úlohy CFLP budou reprezentovat relativní toky zboží mezi zdroji a zákazníky. Cílem úlohy je zvolit nejvhodnější rozmístění zdrojů v lokalitách k tomu určených a přiřadit zákazníky zprovozněným zdrojům tak, aby hodnota celkových nákladů na provoz zprovozněných zdrojů a na zásobování zákazníků a za podmínky nepřekročení kapacity zprovozněných zdrojů byla minimální.

Zdroj: VAN ROY, T. J. [18]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů

J – množina zákazníků

c_{ij} – náklady na splnění požadavku zákazníka $j \in J$ zdrojem zprovozněným v lokalitě $i \in I$

b_j – celkový požadavek zákazníka $j \in J$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

a_i – kapacita zdroje v lokalitě $i \in I$

x_{ij} – podíl požadavku zákazníka $j \in J$, který je uspokojen zdrojem zprovozněným v lokalitě $i \in I$

y_i – bivalentní proměnná reprezentující zprovoznění zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \qquad (19)$$

za podmínek

$$\sum_{j \in J} b_j x_{ij} \leq a_i y_i \qquad \text{pro } i \in I \qquad (20)$$

$$\sum_{i \in I} x_{ij} = 1 \qquad \text{pro } j \in J \qquad (21)$$

$$x_{ij} \geq 0 \qquad \text{pro } i \in I, j \in J \qquad (22)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (23)$$

Funkce (19) je účelovou funkcí modelu a je složená ze dvou částí. První část reprezentuje náklady spojené se zásobováním zákazníka. Druhá část jsou náklady na zprovoznění zdrojů v lokalitách. Skupina podmínek (20) zabezpečuje, že požadavky zákazníků přidělených zdrojům nepřekročí jejich kapacity. Počet podmínek (20) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek zároveň vytváří vazby mezi bivalentními proměnnými v modelu. Skupina podmínek (21) zajistí, že požadavky zákazníků budou splněny. Počet podmínek (21) se rovná počtu zákazníků. Skupiny podmínek (22) a (23) vymezují definiční obory proměnných. Počet podmínek (22) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a zákazníků. Počet podmínek (23) se rovná počtu lokalit vhodných pro umístění zdrojů.

Celkový počet omezujících podmínek v modelu je $2\,|I| + |J| + |I|\,|J|$.

Text programu pro typ lokační úlohy CFLP v programovacím jazyce Mosel má následující tvar:

```
model CFLP
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
c:array(zdroj,zakaznik)of real
x:array(zdroj,zakaznik)of mpvar
f:array(zdroj)of real
y:array(zdroj)of mpvar
b:array(zakaznik)of real
a:array(zdroj)of real
end-declarations
forall(i in zdroj)sum(j in zakaznik)b(j)*x(i,j)<=a(i)*y(i)
forall(j in zakaznik)sum(i in zdroj)x(i,j)=1
forall(i in zdroj)y(i)is_binary
forall(i in zdroj, j in zakaznik)x(i,j)>=0
celkove_naklady:=sum(i in zdroj,j in zakaznik)c(i,j)*x(i,j)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

c::	*[80,95,73,*	*f::*	*[150,165,138]*
	69,82,96,	*b::*	*[200,195,215]*
	85,76,92]	*a::*	*[410,390,450]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 517.125 Kc

x(1,1)=0.975 *x(1,3)=1* *x(3,1)=0.025* *x(3,2)=1* *y(1)=1* *y(3)=1*

3.1.4 Single-source Capacitated Facility Location Problem (SSCFLP)

Model SSCFLP je kapacitní verze nejjednoduššího lokačního modelu, který byl popsán v části 2.1. Tento model se liší od CFL nebo CFLP tím, že celkový požadavek zákazníka musí být uspokojen pouze jedním zdrojem. Cílem modelu je zvolit nejvhodnější lokality pro rozmístění zdrojů a přiřadit zákazníky k těm zdrojům tak, aby celkové náklady na zásobování zákazníků a provoz zdrojů byly minimální.

Zdroj: DARBY – DOWMAN, K. a H. S. LEWIS [7]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů

J – množina zákazníků

b_j – požadavek zákazníka $j \in J$

a_i – kapacita zdroje v lokalitě $i \in I$

c_{ij} – celkové náklady na uspokojení požadavku zákazníka $j \in J$ zdrojem v lokalitě $i \in I$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

x_{ij} – bivalentní proměnná reprezentující přiřazení zákazníka $j \in J$ ke zdroji v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $x_{ij} = 1$, potom je zákazník $j \in J$ přiřazen ke zdroji v lokalitě $i \in I$, pokud je $x_{ij} = 0$, potom zákazník $j \in J$ není přiřazen ke zdroji v lokalitě $i \in I$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \qquad (24)$$

za podmínek

$$\sum_{j \in J} b_j x_{ij} \leq a_i \qquad \text{pro } i \in I \qquad (25)$$

$$\sum_{i \in I} x_{ij} = 1 \qquad \text{pro } j \in J \qquad (26)$$

$$x_{ij} \leq y_i \qquad \text{pro } i \in I, j \in J \qquad (27)$$

$$x_{ij} \in \{0,1\} \qquad \text{pro } i \in I, j \in J \qquad (28)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (29)$$

Funkce (24) je účelovou funkcí modelu a je složená ze dvou částí. První část reprezentuje náklady spojené se zásobováním zákazníka. Druhá část reprezentuje provozní náklady zdrojů v lokalitách. Skupina podmínek (25) zabezpečuje, že celkový požadavek zákazníků uspokojených stejným zdrojem nepřekročí jeho kapacitu. Počet podmínek (25) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek (26) zajistí, že každý zákazník bude přiřazen právě jednomu zdroji. Počet podmínek (26) se rovná počtu zákazníků. Skupina podmínek (27) udává do vzájemného vztahu dvě bivalentní proměnné a zajistí, že když je zákazník přiřazen lokalitě, tak je v lokalitě zprovozněn zdroj a dále, když v lokalitě není zprovozněn zdroj, tak dané lokalitě není přiřazen žádný zákazník. Počet podmínek (27) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a zákazníků. Skupiny podmínek (28) a (29) vymezují definiční obory proměných použitých v modelu. Počet podmínek (28) se rovná součinu počtů zdrojů a zákazníků. Počet podmínek (29) se rovná počtu zdrojů v lokalitách.

Celkový počet omezujících podmínek v modelu je 2 $|I|$ +$|J|$ + 2$|I||J|$.

V literatuře se můžeme ještě setkat s dalšími názvy modelu SSCFLP: Single Source Capacitated Multi Facility Location Problem (Z. Ulukan a E. Demircioglu (2017)).

Text programu pro typ lokační úlohy SSCFLP v programovacím jazyce Mosel má následující tvar:

```
model SSCFLP
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
c:array(zdroj,zakaznik)of real
x:array(zdroj,zakaznik)of mpvar
f:array(zdroj)of real
y:array(zdroj)of mpvar
b:array(zakaznik)of real
a:array(zdroj)of real
end-declarations
forall(i in zdroj)sum(j in zakaznik)b(j)*x(i,j)<=a(i)
forall(j in zakaznik)sum(i in zdroj)x(i,j)=1
forall(i in zdroj, j in zakaznik)x(i,j)<=y(i)
forall(i in zdroj)y(i)is_binary
forall(i in zdroj, j in zakaznik)x(i,j)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik)c(i,j)*x(i,j)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

```
c::  [80,95,73,          f::  [150,165,138]
      69,82,96,          b::  [200,195,215]
      85,76,92]          a::  [410,390,450]
```

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 522 Kc

x(1,3)=1 *x(3,1)=1* *x(3,2)=1* *y(1)=1* *y(3)=1*

3.1.5 Generalized Capacited Facility Location (GCFL)

Model GCFL je obdobou CFLP s tím rozdílem, že každý zdroj v lokalitě definuje nejen horní hranici kapacity, ale i dolní hranici zásobování. To znamená, že pokud předpokládaný přepravní tok ze zdroje bude nižší než jeho stanovená dolní hranice, tak zdroj v dané lokalitě nebude zprovozněn. Cílem úlohy je zvolit nejvhodnější rozmístění zdrojů v lokalitách k tomu určených a přiřadit zákazníky zprovozněným zdrojům tak, aby hodnota celkových nákladů na provoz zprovozněných zdrojů a na zásobování zákazníků byla minimální.

Zdroj: AIKENS, C. [3]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů

J – množina zákazníků

c_{ij} – celkové náklady na uspokojení požadavku zákazníka $j \in J$ zdrojem v lokalitě $i \in I$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

$\underline{V_i}$ – dolní hranice zásobování zdroje v lokalitě $i \in I$

$\bar{V}_i$ – horní hranice zásobování zdroje v lokalitě $i \in I$

b_j – požadavek zákaznika $j \in J$

x_{ij} – podíl požadavku zákazníka $j \in J$, který je uspokojen zdrojem v lokalitě $i \in I$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} + \sum_{i \in I} f_i y_i \tag{30}$$

za podmínek

$$\sum_{j \in J} b_j x_{ij} \leq y_i \bar{V}_i \quad \text{pro } i \in I \tag{31}$$

$$\underline{V_i} y_i \leq \sum_{j \in J} b_j x_{ij} \quad \text{pro } i \in I \tag{32}$$

$$\sum_{i \in I} x_{ij} = 1 \quad \text{pro } j \in J \tag{33}$$

$$x_{ij} \geq 0 \quad \text{pro } i \in I, j \in J \tag{34}$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (35)$$

Funkce (30) je účelovou funkcí modelu a je složená ze dvou částí. První část reprezentuje náklady spojené se zásobováním zákazníka, druhá část reprezentuje náklady na zprovoznění zdrojů v lokalitách. Skupina podmínek (31) zabezpečuje, že pokud je zdroj v lokalitě zprovozněn, tak celkový přepravní tok z něj realizovaný nepřekročí jeho kapacitu. Počet podmínek (31) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek (32) zabezpečuje, že pokud je zdroj v lokalitě zprovozněn, tak tok zboží, který se z něj uskuteční, bude splňovat podmínku minimálního toku kladeného na danou lokalitu. Počet podmínek (32) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek (33) zajistí, že celkový požadavek zákazníka bude uspokojen. Počet podmínek (33) se rovná počtu zákazníků. Skupiny podmínek (34) a (35) vymezují definiční obory proměnných použitých v modelu. Počet podmínek (34) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a zákazníků. Počet podmínek (35) se rovná počtu lokalit vhodných pro umístění zdrojů.

Celkový počet omezujících podmínek v modelu je 3 $|I|$ + $|J|$ + $|I|$ $|J|$

Text programu pro typ lokační úlohy GCFL v programovacím jazyce Mosel má následující tvar:

```
model GCFL
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..2
konstanta=1..2
c:array(zdroj,zakaznik)of real
f:array(zdroj)of real
x:array(zdroj,zakaznik)of mpvar
y:array(zdroj)of mpvar
v1:array(zdroj)of real                    !dolni hranice zasobovani
v2:array(zdroj)of real                    !horni hranice zasobovani
b:array(zakaznik)of real
end-declarations
forall(i in zdroj)v1(i)*y(i)<=sum(j in zakaznik)b(j)*x(i,j)
forall(i in zdroj)v2(i)*y(i)>=sum(j in zakaznik)b(j)*x(i,j)
forall(j in zakaznik)sum( i in zdroj)x(i,j)=1
forall(i in zdroj,j in zakaznik)x(i,j)>=0
forall(i in zdroj)y(i)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik)c(i,j)*x(i,j)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

```
c::  [25,33,          f::   [360,420,180]
      19,42,          v1::  [150,90,110]
      39,50]          v2::  [360,270,250]
                      b::   [290,225]
```

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 605.483 Kc

x(1,1)=0.465517 x(1,2)=1 x(3,1)=0.534483 y(1)=1 y(3)=1

3.2 Víceúrovňové jednokomoditní lokační úlohy

Vzhledem k struktuře logistického řetezce jsou pro logistickou praxi vhodnější úlohy lokačního typu, ve kterých je nutno plánovat rozmístění zdrojů na více úrovních a dál optimalizovat přepravu mezi jednotlivými úrovněmi. Na těchto úrovních mohou být rozmístěna střediska obsluhy, depa, sklady, výrobní závody, centra překladky atd. v závislosti na velikosti řetězce.

Tyto typy úloh jsou v odborné literatuře nazývány např. Hierarchical Facility Location Problem, Two-stage Location Problem Multi Level Uncapacitated Facility Location Problem, Transportation problem, Single-flow hierarchical facility location problem, Two-level Capacitated Facility Location problem, Simple Uncapacitated Multi-echelon Facility Location Problem, Two-stage Capacitated Facility Location Problem. Z uvedené množiny úloh bude v předložené bakalářské práci prezentováno pět úloh označovaných v literatuře jako Simple Uncapacitated Multi-echelon Facility Location Problem, Multi Level Uncapacitated Facility Location Problem, Two-stage Capacitated Facility Location Problem, Two-level Capacitated Facility Location problem, Two-stage Location Problem.

3.2.1 Simple Uncapacitated Multi-echelon Facility Location Problem (SUPW)

V úloze SUPW je definována dopravní síť, v níž je rozmístěna množina zákazníků J, množina lokalit vhodných pro rozmístění zdrojů I na vyšší úrovni a množinu lokalit vhodných pro umístění zdrojů na nižší úrovni K. Aby bylo možno rozlišit lokalitu na vyšší úrovni od lokality na nižší úrovni, budou v lokalitách na vyšší úrovni umísťovány zdroje a v lokalitách na nižší úrovni sklady (v kontextu logistických aplikací). Požadavek zákazníka tedy bude uspokojen zdrojem vždy přes sklad. Vzhledem k tomu, že proměnná x_{ijk}, analogicky jako v úloze SUFL, reprezentuje podíl požadavku, zásobování se může provádět z více než jednoho zdroje a přes více než jeden sklad. Cílem modelu je zvolit nejvhodnější lokality pro rozmístění zdrojů a skladů v síti a přiřadit zákazníky k těm zdrojům a skladům tak, aby náklady na celý proces zásobování dosáhly svého minima.

Zdroj: AIKENS, C. [3]

Symboly v modelu mají následující významy:

I – množina lokalit vhodných pro umístění zdroje

J – množina zákazníků

K – množina lokalit vhodných pro umístění skladu

c_{ijk} – celkové náklady na uspokojení požadavku zákazníka $j \in J$ zdrojem v lokalitě $i \in I$ přes sklad v lokalitě $k \in K$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

g_k – fixní náklady na zprovoznění skladu v lokalitě $k \in K$

x_{ijk} – podíl požadavku zákazníka $j \in J,$ který je uspokojen zdrojem v lokalitě $i \in I$ přes sklad v lokalitě $k \in K$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

z_k – bivalentní proměnná reprezentující provozování skladu v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $z_j = 1$, potom je sklad v lokalitě $k \in K$ v provozu, pokud je $z_j = 0$, potom sklad v lokalitě $k \in K$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\ f(x,y,z) = \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} c_{ijk} x_{ijk} + \sum_{i \in I} f_i y_i + \sum_{k \in K} g_k z_k \qquad (36)$$

za podmínek

$$\sum_{i \in I} \sum_{k \in K} x_{ijk} = 1 \qquad \text{pro } j \in J \qquad (37)$$

$$\sum_{j \in J} \sum_{k \in K} x_{ijk} \leq y_i \qquad \text{pro } i \in I \qquad (38)$$

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} \leq z_k \qquad \text{pro } k \in K \qquad (39)$$

$$x_{ijk} \geq 0 \qquad \text{pro } i \in I, j \in J, k \in K \qquad (40)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (41)$$

$$z_k \in \{0,1\} \qquad \text{pro } k \in K \qquad (42)$$

Funkce (36) je účelovou funkcí modelu složenou ze tří částí. První část reprezentuje náklady spojené s přiřazením zákazníků ke zdrojům a skladům v lokalitách. Druhá část reprezentuje náklady spojené se zprovozněním zdrojů a třetí část reprezentuje náklady spojené se zprovozněním skladů v lokalitách. Skupina podmínek (37) zabezpečuje, že celkový požadavek každého zákazníka bude uspokojen. Počet podmínek (37) se rovná počtu zákazníků. Skupiny podmínek (38) a (39) zajišťují logické vazby mezi proměnnými, tzn., že když bude některý ze zákazníků zásobován z některého zdroje/prostřednictvím některého skladu, budou tyto lokality v provozu a pokud nebudou zdroje/sklady v těchto lokalitách v provozu, nebude z nich zásobován žádný zákazník. Počet podmínek (38) se rovná počtu lokalit vhodných pro umístění zdrojů, počet podmínek (39) se rovná počtu lokalit vhodných pro umístění skladů. Skupiny podmínek (40), (41) a (42) vymezují definiční obory proměnných. Počet podmínek (40) se rovná počtu relací mezi zdroji, sklady a zákazníky, počet podmínek (41) se rovná počtu lokalit vhodných pro umístění zdrojů a počet podmínek (42) se rovná počtu lokalit vhodných pro umístění skladů.

Celkový počet omezujících podmínek v modelu je $2|I| + |J| + 2|K| + |I||J||K|$.

Text programu pro typ lokační úlohy SUPW v programovacím jazyce Mosel má následující tvar:

```
model SUPW
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..2
sklad=1..3
c:array(zdroj,zakaznik,sklad)of real
x:array(zdroj,zakaznik,sklad)of mpvar
f:array(zdroj)of real
y:array(zdroj)of mpvar
g:array(sklad)of real
z:array(sklad)of mpvar
end-declarations
forall(j in zakaznik)sum(i in zdroj, k in sklad)x(i,j,k)=1
forall(i in zdroj)sum(k in sklad, j in zakaznik)x(i,j,k)<=y(i)
forall(k in sklad)sum(i in zdroj,j in zakaznik)x(i,j,k)<=z(k)
forall(i in zdroj, j in zakaznik, k in sklad)x(i,j,k)>=0
forall(i in zdroj)y(i)is_binary
forall(k in sklad)z(k)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik, k in sklad)c(i,j,k)*x(i,j,k)+sum(i in zdroj)f(i)*y(i) + sum(k
in sklad)g(k)*z(k)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik, k in sklad)writeln("x(",i,",",j,",",k,")=",getsol(x(i,j,k)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
forall(k in sklad)writeln("z(",k,")=",getsol(z(k)))
end-model
```

Pro hodnoty konstant:

c::	*[220,250,280,190,305,210, 300,210,150,225,180,230, 250,265,195,300,270,200]*	*f::*	*[300,300,275]*	*g::*	*[270,280,305]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 1515 Kc

x(1,2,1)=1 *x(2,1,3)=1* *y(1)=1* *y(2)=1* *z(1)=1* *z(3)=1*

3.2.2 Multi Level Uncapacitated Facility Location Problem (MLUFLP)

V úloze MLUFLP je definována dopravní síť, která obsahuje množinu zákazníků a dvě množiny lokalit reprezentujících rozmístění zdrojů ve dvou úrovních. To znamená, že MLUFLP je víceúrovňová úloha, kde se tok komodity pohybuje nejdříve ze zdroje vyšší úrovně (zdroj úrovně 2, v předchozí víceúrovňové lokační úloze by tento typ označen jako zdroj) do zdroje nižší úrovně (zdroj úrovně 1, v předchozí víceúrovňové lokační úloze by tento typ označen jako sklad) a následně potom ze zdroje nižší úrovně k zákazníkovi. Kromě toho jsou dopředu definovány počty zdrojů jednotlivých úrovní, které mají být zprovozněny. Úlohou je zvolit nejvhodnější rozmístění zdrojů v obou úrovních v síti a rozhodnout o zásobování zákazníků z nich tak, aby celkové náklady na přiřazení zákazníků ke zdrojům a náklady na zprovoznění zdrojů byly minimální.

Zdroj: KRATICA, J., D. DUGOŠIJA a A. SAVIĆ [11]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů úrovně 2

J – množina zákazníků

K – množina potenciálních lokalit vhodných pro umístění zdrojů úrovně 1

w_{ijk} – náklady plynoucí z přiřazení zákazníka $j \in J$ ke zdroji úrovně 1 v lokalitě $k \in K$ a ke zdroji úrovně 2 v lokalitě $i \in I$

p – počet zdrojů úrovně 1, které mají být zřízeny

f_k – náklady na zprovoznění zdroje úrovně 1 v lokalitě $k \in K$

q – počet zdrojů úrovně 2, které mají být zřízeny

g_i – náklady na zprovoznění zdroje úrovně 2 v lokalitě $i \in I$

x_{ijk} – bivalentní proměnná reprezentující přiřazení zákazníka $j \in J$ ke zdroji úrovně 2 v lokalitě $i \in I$ přes zdroj úrovně 1 v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $x_{ijk} = 1$, potom zákazník $j \in J$ bude zásobován ze zdroje úrovně 2 v lokalitě $i \in I$ přes zdroj úrovně 1 v lokalitě $k \in K$, pokud je $x_{ijk} = 0$, potom zákazník $j \in J$ nebude zásobován ze zdroje úrovně 2 v lokalitě $i \in I$ přes zdroj úrovně 1 v lokalitě $k \in K$

y_k – bivalentní proměnná reprezentující provozování zdroje úrovně 1 v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $y_k = 1$, potom je zdroj úrovně 1 v lokalitě $k \in K$ v provozu, pokud je $y_k = 0$, potom zdroj úrovně 1 v lokalitě $k \in K$ v provozu není

z_i – bivalentní proměnná reprezentující provozování zdroje úrovně 2 v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $z_i = 1$, potom je zdroj úrovně 2 v lokalitě $i \in I$ v provozu, pokud je $z_i = 0$, potom zdroj úrovně 2 v lokalitě $i \in I$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\ f(x,y) = \sum_{i \in I} \sum_{k \in K} \sum_{j \in J} x_{ijk} w_{ijk} + \sum_{i \in I} g_i z_i + \sum_{k \in k} f_k y_k \tag{43}$$

za podmínek

$$\sum_{k \in K} y_k = p \qquad (44)$$

$$\sum_{i \in I} z_i = q \qquad (45)$$

$$\sum_{i \in I} \sum_{k \in K} x_{ijk} = 1 \qquad \text{pro } j \in J \qquad (46)$$

$$x_{ijk} \leq y_k \qquad \text{pro } i \in I, j \in J, k \in K \qquad (47)$$

$$x_{ijk} \leq z_i \qquad \text{pro } i \in I, j \in J, k \in K \qquad (48)$$

$$x_{ijk} \in \{0,1\} \qquad \text{pro } i \in I, j \in J, k \in K \qquad (49)$$

$$y_k \in \{0,1\} \qquad \text{pro } k \in K \qquad (50)$$

$$z_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (51)$$

Funkce (43) je účelovou funkcí modelu, která je složena ze tří částí. První část reprezentuje náklady spojené s přiřazením zákazníků ke zdrojům úrovně 2 přes zdroje úrovně 1 v lokalitách. Zbývající dvě části reprezentují náklady na zprovoznění zdrojů úrovně 1 a 2 v lokalitách. Skupiny podmínek (44) a (45) zabezpečují, že počty zprovozněných zdrojů úrovně 1 a 2 se budou rovnat předem definovaným hodnotám. V modelu je vždy pouze jedna podmínka (44) a jedna podmínka (45). Skupina podmínek (46) zajistí, že každý zákazník bude zásobován z právě jednoho zdroje úrovně 2 přes právě jeden zdroj úrovně 1. Počet podmínek (46) se rovná počtu zákazníků. Skupiny podmínek (47) a (48) plní úlohu vazebních podmínek. Tzn., že když dojde k zásobování zákazníka ze zdroje úrovně 2 přes zdroj v úrovni 1, potom zdroje v obou úrovních budou zprovozněny a když nebude některý ze zdrojů v obou úrovních v provozu, potom k zásobování zákazníka po definované trase nedojde. Počty podmínek v každé ze skupin (47) a (48) se rovnají součinům počtů lokalit vhodných pro umístění zdrojů úrovně 1, úrovně 2 a zákazníků. Skupiny podmínek (49), (50) a (51) reprezentují definiční obory proměnných. Počet podmínek (49) se rovná součinu počtů lokalit vhodných pro umístění zdrojů úrovně 1, úrovně 2 a zákazníků. Počet podmínek (50) odpovídá počtu lokalit vhodných pro umístění zdroje úrovně 1 a počet podmínek (51) odpovídá počtu lokalit vhodných pro umístění zdrojů úrovně 2.

Celkový počet omezujících podmínek v modelu je 2 + $|I|$ + $|J|$ + $|K|$ + $3|I||J||K|$

Text programu pro typ lokační úlohy MLUFLP v programovacím jazyce Mosel má následující tvar:

```
model MLUFLP
uses "mmxprs"
declarations
zdroj2=1..3
zakaznik=1..2
zdroj1=1..3
x:array(zdroj2,zakaznik,zdroj1)of mpvar
w:array(zdroj2,zakaznik,zdroj1)of real
g:array(zdroj2)of real
z:array(zdroj2)of mpvar
f:array(zdroj1)of real
y:array(zdroj1)of mpvar
p:real
```

```
q:real
end-declarations
sum(k in zdroj1)y(k)=p
sum(i in zdroj2)z(i)=q
forall(j in zakaznik)sum(i in zdroj2,k in zdroj1)x(i,j,k)=1
forall(i in zdroj2,k in zdroj1,j in zakaznik)x(i,j,k)<=y(k)
forall(i in zdroj2,k in zdroj1,j in zakaznik)x(i,j,k)<=z(i)
forall(i in zdroj2,k in zdroj1,j in zakaznik)x(i,j,k) is_binary
forall(k in zdroj1)y(k)is_binary
forall(i in zdroj2)z(i)is_binary
celkove_naklady:=sum(k in zdroj1, j in zakaznik, i in zdroj2)x(i,j,k)*w(i,j,k)+sum(i in zdroj2)
g(i)*z(i)+sum(k in zdroj1)f(k)*y(k)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj2,j in zakaznik, k in zdroj1)writeln("x(",i,",",j,",",k,")=",getsol(x(i,j,k)))
forall(i in zdroj2)writeln("z(",i,")=",getsol(z(i)))
forall(k in zdroj1)writeln("y(",k,")=",getsol(y(k)))
end-model
```

Pro hodnoty konstant:

w::	*[20,15,38,25,16,39, 39,11,26,17,42,45, 19,43,48,39,10,13]*	*g::*	*[150,280,310]*	*p:=*	*2*
		f::	*[420,330,250]*	*q:=*	*2*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 1037 Kc

x(1,2,2)=1 x(2,1,2)=1 z(1)=1 z(2)=1 y(2)=1 y(3)=1

3.2.3 Two-stage Capacitated Facility Location Problem (TSCFLP)

Model TSCFLP je rozšíření CFLP v tom smyslu, že mezi zákazníkem a zdrojem ještě bude umístěn jeden prvek dodavatelského řetězce nazývaný např. depo. V daném modelu je rozmístění zdrojů předem dáno a rozhoduje se o zprovoznění dep. Cílem modelu je zvolit nejvhodnější lokality pro rozmístění dep a přiřadit zákazníky zprovozněným depům tak, aby se celkové náklady na zásobování zákazníků a na zprovoznění dep v lokalitách minimalizovaly.

Zdroj: KLOSE, A. [9]

Symboly v modelu mají následující významy:

I – množina zdrojů

J – množina zákazníků

K – množina lokalit vhodných pro umístění depa

a_i – výrobní kapacita zdroje $i \in I$

m_k – kapacita depa v lokalitě $k \in K$

b_j – požadavek zákazníka $j \in J$

B – celkový požadavek všech zákazníků v úloze

t_{ik} – jednotkové přepravní náklady ze zdroje $i \in I$ do depa v lokalitě $k \in K$

c_{jk} – celkové přepravní náklady na zásobování zákazníka $j \in J$ z depa umístěného v lokalitě $k \in K$

f_k – fixní náklady na zprovoznění depa v lokalitě $k \in K$

v_{ik} – tok komodit ze zdroje $i \in I$ do depa v lokalitě $k \in K$

y_k – bivalentní proměnná reprezentující provozování depa v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $y_k = 1$, potom je depo v lokalitě $k \in K$ v provozu, pokud je $y_k = 0$, potom depo v lokalitě $k \in K$ v provozu není

x_{jk} – bivalentní proměnná reprezentující přiřazení zákazníka $j \in J$ k depu v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $x_{jk} = 1$, potom je zákazník $j \in J$ přiřazen k depu v lokalitě $k \in K$, pokud je $x_{jk} = 0$, potom zákazník $j \in J$ není přiřazen k depu v lokalitě $k \in K$

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y,v) = \sum_{i \in I} \sum_{k \in K} t_{ik} v_{ik} + \sum_{k \in K} \sum_{j \in J} c_{jk} x_{jk} + \sum_{k \in K} f_k y_k \qquad (52)$$

za podmínek

$$\sum_{k \in k} x_{jk} = 1 \qquad \text{pro } j \in J \qquad (53)$$

$$\sum_{j \in J} b_j x_{jk} \leq m_k y_k \qquad \text{pro } k \in K \qquad (54)$$

$$x_{jk} \leq y_k \qquad \text{pro } k \in K, j \in J \qquad (55)$$

$$\sum_{k \in K} m_k y_k \geq B \qquad (56)$$

$$\sum_{k \in k} v_{ik} \leq a_i \qquad \text{pro } i \in I \qquad (57)$$

$$\sum_{i \in I} v_{ik} = \sum_{j \in J} b_j\, x_{jk} \qquad \text{pro } k \in K \qquad (58)$$

$$v_{ik} \geq 0 \qquad \text{pro } k \in K, i \in I \qquad (59)$$

$$y_k \in \{0,1\} \qquad \text{pro } k \in K \qquad (60)$$

$$x_{jk} \in \{0,1\} \qquad \text{pro } k \in K, j \in J \qquad (61)$$

Funkce (52) je účelovou funkcí modelu a je složená ze tří částí. První část reprezentuje přepravní náklady, které vznikají mezi existujícími zdroji a jednotlivými lokalitami vhodnými pro umístění dep. Druhá část reprezentuje náklady spojené s zasobováním zákazníků ze zprovozněných dep. Třetí část reprezentuje náklady na zprovoznění dep v lokalitách. Skupina podmínek (53) zajistí, že každý zákazník bude přiřazen právě jednomu zprovozněnému depu. Počet podmínek (53) se rovná počtu zákazníků. Skupina podmínek (54) zabezpečuje, že pokud je depo zprovozněno, tak celkový požadavek přiřazených zákazníků nepřekročí jeho kapacitu. Počet podmínek (54) se rovná počtu lokalit vhodných pro umístění dep. Skupina podmínek (55) udává do vzájemného vztahu dvě bivalentní proměnné a zajistí, že když zákazník bude přidělen lokalitě, tak v dané lokalitě bude zřízeno depo a když v dané lokalitě nebude zřízeno depo, potom lokalitě nebude přiřazen žádný zákazník. Počet podmínek (55) se rovná součinu počtů lokalit vhodných pro umístění dep a zákazniků. Skupina podmínek (56) zajistí, že celková kapacita dep bude dostatečná a taková podmínka bude jen jedna. Skupina podmínek (57) zajišťuje, že celkový tok mezi zdrojem a depy nepřekročí výrobní kapacitu zdroje. Počet podmínek (57) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek (58) zajistí kontinuitu toků zboží v depech. Počet podmínek (58) se rovná počtu lokalit vhodných pro umístění depa. Skupiny podmínek (59), (60) a (61) vymezují definiční obory proměných použitých v modelu. Počet podmínek (59) se rovná součinu počtů zdrojů a lokalit vhodných pro umístění dep, počet podmínek (60) se rovná počtu lokalit vhodných pro umístění dep. Počet podmínek (61) se rovná součinu počtů lokalit vhodných pro umístění dep a zákazníků.

Celkový počet omezujících podmínek v modelu je $|I| + 2|J| + 3|K| + 2|K||J| + |K||I|$

Text programu pro typ lokační úlohy TSCFLP v programovacím jazyce Mosel má následující tvar:

```
model TSCFLP
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
depo=1..3
t:array(zdroj,depo)of real
v:array(zdroj,depo)of mpvar
c:array(zakaznik,depo)of real
x:array(zakaznik,depo)of mpvar
f:array(depo)of real
y:array(depo)of mpvar
b:array(zakaznik)of real
m:array(depo)of real
a:array(zdroj)of real
B:real
end-declarations
B:=sum(j in zakaznik)b(j)
forall(j in zakaznik)sum(k in depo)x(j,k)=1
forall(k in depo)sum(j in zakaznik)b(j)*x(j,k)<=m(k)*y(k)
forall(k in depo, j in zakaznik)x(j,k)<=y(k)
forall(j in zakaznik)sum(k in depo)m(k)*y(k)>=B
forall(i in zdroj)sum(k in depo)v(i,k)<=a(i)
forall(k in depo)sum(i in zdroj)v(i,k)=sum(j in zakaznik)b(j)*x(j,k)
forall(k in depo, i in zdroj)v(i,k)>=0
forall(k in depo)y(k)is_binary
forall(k in depo,j in zakaznik)x(j,k)is_binary
celkove_naklady:=sum(i in zdroj,k in depo)t(i,k)*v(i,k) + sum(j in zakaznik, k in depo)c(j,k)*x(j,k)
+sum(k in depo)f(k)*y(k)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
```

```
forall(k in depo,j in zakaznik)writeln("x(",j,",",k,")=",getsol(x(j,k)))
forall(k in depo,i in zdroj)writeln("v(",i,",",k,")=",getsol(v(i,k)))
forall(k in depo)writeln("y(",k,")=",getsol(y(k)))
end-model
```

Pro hodnoty konstant:

f::	*[420,380,495]*	t::	*[20,15,23, 16,18,27, 22,24,11]*	c:: *[33,28,39, 25,29,41, 32,27,24]*
b::	*[360,425,398]*			
m::	*[890,780,960]*			
a::	*[920,730,855]*			

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 15421 Kc

x(1,2)=1 x(2,3)=1 x(3,3)=1 v(1,2)=360 v(3,3)=823 y(2)=1 y(3)=1

3.2.4 Two-level Capacitated Facility Location problem (TLCFLP)

V úloze TLCFLP je definována dopravní síť, která obsahuje množinu zákazníků a dvě množiny lokalit vhodných pro umístění zdrojů ve dvou úrovních. Oproti předchozím víceúrovňovým modelům se v úloze rozmísťují nejenom zdroje na nižší úrovni (v předchozí úloze označené jako depa) ale i na vyšší úrovni. Zákazník v tomto modelu může mít přímé spojení jen se zdrojem na nižší úrovni. Cílem úlohy je zvolit nejvhodnější lokality pro rozmístění zdrojů v obou úrovních a přiřadit zákazníky ke zdrojům na nižší úrovni tak, aby celkové náklady na zasobování zákazníků a na zprovoznění zdrojů na obou úrovních byly minimální.

Zdroj: KLOSE, A. a A. DREXL [10]

Symboly v modelu mají následující významy:

I – množina lokalit vhodných pro umístění zdrojů vyšší úrovně

J – množina zákazníků

K – množina potenciálních lokalit vhodných pro umístění zdrojů nižší úrovně

b_j – požadavek zákazníka $j \in J$

B – celkový požadavek všech zákazníků $j \in J$

f_i – fixní náklady na zprovoznění zdroje vyšší úrovně v lokalitě $i \in I$

a_i – kapacita zdroje vyšší úrovně v lokalitě $i \in I$

t_{ik} – jednotkové přepravní náklady z lokality $i \in I$ do lokality $k \in K$

c_{kj} – celkové přepravní náklady z lokality $k \in K$ k zákazníkovi $j \in J$

s_k – kapacita zdroje nižší úrovně v lokalitě $k \in K$

g_k – fixní náklady na zprovoznění zdroje nižší úrovně v lokalitě $k \in K$ (v dané úloze zprovoznění zdrojů nižší úrovně nevyvolavá žádné náklady a proto $g_k = 0$ pro $k \in K$)

v_{ik} – tok komodity ze zdroje vyšší úrovně v lokalitě $i \in I$ do zdroje nižší úrovně v lokalitě $k \in K$

z_i – bivalentní proměnná reprezentující provozování zdroje vyšší úrovně v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $\gamma_i = 1$, potom je zdroj vyšší úrovně v lokalitě $i \in I$ v provozu, pokud je $\gamma_i = 0$, potom zdroje vyšší úrovně v lokalitě $i \in I$ v provozu není

x_{kj} – bivalentní proměnná reprezentující přiřazení zákazníka $j \in J$ ke zdroji nižší úrovně v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $x_{kj} = 1$, potom je zákazník $j \in J$ přiřazen zdroji nižší úrovně v lokalitě $k \in K$, pokud je $x_{kj} = 0$, potom zákazník $j \in J$ není přiřazen ke zdroji nižší úrovně v lokalitě $k \in K$

y_k – bivalentní proměnná reprezentující provozování zdroje nižší úrovně v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $y_k = 1$, potom je zdroj nižší úrovně v lokalitě $k \in K$ v provozu, pokud je $y_k = 0$, potom zdroj nižší úrovně v lokalitě $k \in K$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y,v) = \sum_{i \in I} \sum_{k \in k} t_{ik} v_{ik} + \sum_{k \in K} \sum_{j \in J} c_{kj} x_{kj} + \sum_{i \in I} f_i z_i \qquad (62)$$

za podmínek

$$\sum_{k \in k} x_{kj} = 1 \qquad \text{pro } j \in J \qquad (63)$$

$$x_{kj} \leq y_k \qquad \text{pro } k \in K, j \in J \qquad (64)$$

$$\sum_{k \in K} s_k y_k \geq B \qquad (65)$$

$$\sum_{j \in J} x_{kj}\, b_j \leq s_k y_k \qquad \text{pro } k \in K \qquad (66)$$

$$\sum_{i \in I} v_{ik} = \sum_{j \in J} x_{kj}\, b_j \qquad \text{pro } k \in K \qquad (67)$$

$$\sum_{i \in I} a_i z_i \geq B \qquad (68)$$

$$\sum_{k \in K} v_{ik} \leq a_i z_i \qquad \text{pro } i \in I \qquad (69)$$

$$v_{ik} \leq min\{s_k, a_i\} z_i \qquad \text{pro } i \in I,\ k \in K \qquad (70)$$

$$y_k \in \{0,1\} \qquad \text{pro } k \in K \qquad (71)$$

$$x_{kj} \in \{0,1\} \qquad \text{pro } k \in K, j \in J \qquad (72)$$

$$v_{ik} \geq 0 \qquad \text{pro } i \in I, k \in K \qquad (73)$$

$$z_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (74)$$

Funkce (62) je účelovou funkcí modelu a je složená ze tří částí. První část reprezentuje náklady, které vznikají mezi zdroji vyšší úrovně a zdroji nižší úrovně. Druhá část reprezentuje náklady spojené se zásobováním zákazníků. Třetí část reprezentuje náklady na zprovoznění zdrojů vyšší úrovně. Skupina podmínek (63) zajistí, že každý zákazník bude přiřazen právě jednomu zdroji nižší úrovně. Počet podmínek (63) se rovná počtu zákazníků. Skupina podmínek (64) vytváří vzajemný vztah mezi kooperujícími bivalentními proměnnými a zajišťuje, že když bude zákazník přiřazen lokalitě pro zdroj nižší úrovně, potom v dané lokalitě bude zdroj nižší úrovně v provozu a dále, že pokud zdroj nižší úrovně v dané lokalitě v provozu nebude, potom jí nebude přidělen žádný zákazník. Počet podmínek (64) se rovná součinu počtů lokalit vhodných pro umístění zdrojů nižší úrovně a zákazníků. Skupina podmínek (65) zajistí, že celková kapacita zdrojů nižší úrovně bude dostatečná. Počet podmínek (65) je roven jedné. Skupina podmínek (66) zabezpečuje, že pokud je zdroj nižší úrovně zprovozněn, tak celkový požadavek uspokojený tímto zdrojem nepřekročí jeho kapacitu. Počet podmínek (66) se rovná počtu lokalit vhodných pro umístění zdrojů nižší úrovně. Skupina podmínek (67) zajišťuje kontinuitu toků ve zdrojích nižší úrovně. Počet podmínek (67) se rovná počtu lokalit vhodných pro umístění zdrojů nižší úrovně. Skupina podmínek (68) zaručuje, že součet kapacit zdrojů vyšší úrovně bude dostatečný. Počet podmínek (68) v modelu je roven 1. Skupina podmínek (69) zajišťuje, že tok zboží ze zdrojů vyšší úrovně nepřekročí jejich kapacity. Počet podmínek (69) se rovná počtu zdrojů vyšší úrovně. Skupina podmínek (70) uvádí, že tok mezi zdroji vyšší a nižší úrovně vyhoví kapacitním omezením obou zdrojů. Počet podmínek (70) se rovná součinu počtů lokalit vhodných pro umístění zdrojů vyšší úrovně a zdrojů nižší úrovně. Skupiny podmínek (71), (72), (73) a (74) vymezují definiční obory proměných použitých v modelu. Počet podmínek (71) se rovná počtu lokalit vhodných pro umístění zdrojů nižší úrovně. Počet podmínek (72) se rovná součinu počtů lokalit vhodných pro umístění zdrojů nižší úrovně a zakázníků. Počet podmínek (73) se rovná součinu počtů lokalit vhodných pro umístění zdrojů vyšší úrovně a zdrojů nižší úrovně. Počet podmínek (74) se rovná počtu lokalit vhodných pro umístění zdrojů vyšší úrovně.

Celkový počet omezujících podmínek v modelu je $2 + 2|I| + |J| + 3|K| + 2|I||K| + 2|J||K|$.

Text programu pro typ lokační úlohy TLCFLP v programovacím jazyce Mosel má následující tvar:

```
model TLCFLP
uses "mmxprs"
declarations
vzdroj=1..3!množina lokalit vhodných pro umístění zdrojů vyšší urovně
nzdroj=1..3!množina lokalit vhodných pro umístění zdrojů nižší úrovně
zakaznik=1..3
t:array(vzdroj,nzdroj)of real
v:array(vzdroj,nzdroj)of mpvar
y:array(nzdroj)of mpvar
f:array(vzdroj)of real
z:array(vzdroj)of mpvar
c:array(nzdroj,zakaznik)of real
x:array(nzdroj,zakaznik)of mpvar
s:array(nzdroj)of real
b:array(zakaznik)of real
a:array(vzdroj)of real
B:real
end-declarations
B:=sum(j in zakaznik)b(j)
forall(j in zakaznik)sum(k in nzdroj)x(k,j)=1
```

```
forall(k in nzdroj, j in zakaznik)x(k,j)<=y(k)
sum(k in nzdroj)s(k)*y(k)>=B
forall(k in nzdroj)sum(j in zakaznik)b(j)*x(k,j)<=s(k)*y(k)
forall(k in nzdroj)sum(i in vzdroj)v(i,k)=sum(j in zakaznik)x(k,j)*b(j)
sum(i in vzdroj)a(i)*z(i)>=B
forall(i in vzdroj)sum(k in nzdroj)v(i,k)<=a(i)*z(i)
forall(k in nzdroj, i in vzdroj) do
if s(k)<a(i) then v(i,k)<=s(k)*z(i)
else v(i,k)<=a(i)*z(i)
end-if
end-do
forall(k in nzdroj)y(k)is_binary
forall(i in vzdroj)z(i)is_binary
forall(k in nzdroj, j in zakaznik)x(k,j)is_binary
forall(i in vzdroj, k in nzdroj)v(i,k)>=0
celkove_naklady:=sum(i in vzdroj,k in nzdroj)t(i,k)*v(i,k)+sum(j in zakaznik,k in nzdroj) c(k,j)*x(k,j)
+sum(i in vzdroj)f(i)*z(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(k in nzdroj,j in zakaznik)writeln("x(",k,",",j,")=",getsol(x(k,j)))
forall(k in nzdroj,i in vzdroj)writeln("v(",i,",",k,")=",getsol(v(i,k)))
forall(k in nzdroj)writeln("y(",k,")=",getsol(y(k)))
forall(i in vzdroj)writeln("z(",i,")=",getsol(z(i)))
end-model
```

Pro hodnoty konstant:

t::	*[40,48,35,*	*c::*	*[32,16,24,*	*f::*	*[620,530,710]*
	29,33,42,		*41,39,17,*	*s::*	*[810,920,780]*
	53,27,32]		*18,19,26]*	*b::*	*[520,330,455]*
				a::	*[930,620,690]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 38173 Kc

x(1,1)=1	*v(2,1)=520*	*y(1)=1*	*z(2)=1*
x(2,2)=1	*v(2,2)=95*	*y(2)=1*	*z(3)=1*
x(2,3)=1	*v(3,2)=690*	*y(3)=1*	

3.2.5 Two-stage Location Problem (GTLP2)

V úloze GTLP2 je definována dopravní síť, která obsahuje množinu zákazníků, množinu lokalit vhodných pro umístění zdrojů a množinu lokalit na depa. Každé depo a zdroj má svou maximalní kapacitu, ale není předem dané jejich rozmístění spolu s počtem, který je třeba rozmístit. Cílem úlohy je určit nejvhodnější lokality pro rozmístění zdrojů a dep v lokalitách tak, aby celkové náklady na zásobování zákazníků a na zprovoznění zdrojů a dep byly minimální.

Rozdíl mezi úlohou GTLP2 a úlohou TLCFLP spočívá v tom, že zatímco v úloze TLCFLP může být zákazník zásobován pouze z jednoho zdroje nižší úrovně a náklady na zprovoznění zdrojů nižší úrovně jsou nulové, v úloze GTLP2 může být zákazník zásobován současně z více zdrojů nižší úrovně (v podmínkách úlohy GTLP2 je hovořeno o depech) a náklady na zprovoznění zdrojů nižší úrovně (dep) mohou být nenulové.

Zdroj: MARÍN, A. a B. PELEGRÍN [13]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů

J – množina zákazníků

K – množina potenciálních lokalit vhodných pro umístění dep

a_i – kapacita zdroje v lokalitě $i \in I$

f_i – instalační náklady zdroj v lokalitě $i \in I$

q_k – kapacita depa v lokalitě $k \in K$

g_k – instalační náklady depa v lokalitě $k \in K$

b_j – požadavek zákazníka $j \in J$

c_{kj} – náklady na zásobování zákazníka $j \in J$ z depa v lokalitě $k \in K$

t_{ki} – přepravní náklady ze zdroje v lokalitě $i \in I$ do depa umístěného v lokalitě $k \in K$

x_{kj} – podíl požadavku zákazníka $j \in J$, který je uspokojen depem umístěným v lokalitě $k \in K$

z_{ki} – podíl produkce zdroje v lokalitě $i \in I$, který je přepraven do depa umístěného v lokalitě $k \in K$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

s_k – bivalentní proměnná reprezentující provozování depa v lokalitě $k \in K$, pokud je po ukončení optimalizačního výpočtu $s_k = 1$, potom je depo v lokalitě $k \in K$ v provozu, pokud je $s_k = 0$, potom depo v lokalitě $k \in K$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y,z,s) = \sum_{k \in K} g_k s_k + \sum_{i \in I} f_i y_i + \sum_{k \in K} \sum_{j \in J} c_{kj} x_{kj} + \sum_{k \in K} \sum_{i \in I} t_{ki} z_{ki} \qquad (75)$$

za podmínek

$$\sum_{k \in K} x_{kj} = 1 \qquad \text{pro } j \in J \qquad (76)$$

$$\sum_{j \in J} b_j x_{kj} = \sum_{i \in I} a_i z_{ki} \qquad \text{pro } k \in K \qquad (77)$$

$$x_{kj} \le s_k \qquad \text{pro } k \in K, j \in J \qquad (78)$$

$$z_{ki} \le y_i \qquad \text{pro } k \in K, i \in I \qquad (79)$$

$$\sum_{k\in K} z_{ki} \leq 1 \qquad \text{pro } i \in I \qquad (80)$$

$$\sum_{i\in I} a_i z_{ki} \leq q_k \qquad \text{pro } k \in K \qquad (81)$$

$$s_k \in \{0,1\} \qquad \text{pro } k \in K \qquad (82)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (83)$$

$$0 \leq x_{kj} \leq 1 \qquad \text{pro } k \in K, j \in J \qquad (84)$$

$$0 \leq z_{ki} \leq 1 \qquad \text{pro } k \in K, i \in I \qquad (85)$$

Funkce (75) je účelovou funkcí modelu a je složena ze čtyř částí. První část reprezentuje náklady na zprovoznění dep v lokalitách. Druhá část reprezentuje náklady na zprovoznění zdrojů v lokalitách. Třetí část reprezentuje náklady spojené se zásobováním zákazníků. Čtvrtá část reprezentuje náklady, které vznikají při přepravě ze zdrojů do dep. Skupina podmínek (76) zajistí, že celkový požadavek každého zákazníka bude splněn. Počet podmínek (76) se rovná počtu zákazníků. Skupina podmínek (77) zabezpečuje kontinuitu toků zboží v depech. Počet podmínek (77) se rovná počtu lokalit vhodných pro umístění dep. Skupina podmínek (78) udává, že zákazník bude zásobován jen ze zprovozněného depa. Počet podmínek (78) se rovná součinu počtů lokalit vhodných pro umístění dep a zákazníků. Skupina podmínek (79) zajistí, že depa bude zásobováno jen ze zprovozněných zdrojů. Počet podmínek (79) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a depa. Skupina podmínek (80) zajistí, že kapacity zdrojů nebudou překročeny. Počet podmínek (80) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek (81) zajistí, že kapacity dep nebudou překročeny. Počet podmínek (81) se rovná počtu lokalit vhodných pro umístění dep.

Skupiny podmínek (82), (83), (84) a (85) vymezují definiční obory proměnných. Počet podmínek (82) se rovná počtu lokalit vhodných pro umístění dep. Počet podmínek (83) se rovná počtu lokalit vhodných pro umístění zdrojů. Počet podmínek (84) se rovná součinu počtů zákazníků a lokalit vhodných pro umístění dep. Počet podmínek (85) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a dep.

Celkový počet omezujících podmínek v modelu je $2|I| + |J| + 3|K| + 2|I||K| + 2|J||K||$.

Text programu pro typ lokační úlohy GTLP2 v programovacím jazyce Mosel má následující tvar:

```
model GTLP2
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
depo=1..3
g:array(depo)of real
s:array(depo)of mpvar
f:array(zdroj)of real
y:array(zdroj)of mpvar
c:array(depo,zakaznik)of real
x:array(depo,zakaznik)of mpvar
t:array(depo,zdroj)of real
```

```
z:array(depo,zdroj)of mpvar
b:array(zakaznik)of real
a:array(zdroj)of real
q:array(depo)of real
end-declarations
forall(j in zakaznik)sum(k in depo)x(k,j)=1
forall(k in depo)sum(j in zakaznik)b(j)*x(k,j)=sum(i in zdroj)a(i)*z(k,i)
forall(k in depo, j in zakaznik)x(k,j)<=s(k)
forall(k in depo,i in zdroj)z(k,i)<=y(i)
forall(i in zdroj)sum(k in depo)z(k,i)<=1
forall(k in depo)sum(i in zdroj)a(i)*z(k,i)<=q(k)
forall(k in depo)s(k)is_binary
forall(i in zdroj)y(i)is_binary
forall(k in depo,j in zakaznik)x(k,j)>=0
forall(k in depo, j in zakaznik)x(k,j)<=1
forall(k in depo, i in zdroj)z(k,i)>=0
forall(k in depo, i in zdroj)z(k,i)<=1
celkove_naklady:=sum(k in depo,j in zakaznik)c(k,j)*x(k,j)+sum(k in depo,i in zdroj) t(k,i)*z(k,i) +
sum(i in zdroj)f(i)*y(i)+sum(k in depo)g(k)*s(k)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(k in depo,j in zakaznik)writeln("x(",k,",",j,")=",getsol(x(k,j)))
forall(k in depo)writeln("s(",k,")=",getsol(s(k)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
forall(k in depo,i in zdroj)writeln("z(",k,",",i,")=",getsol(z(k,i)))
end-model
```

Pro hodnoty konstant:

c::	*[80,95,73, 69,82,96, 85,76,92]*	t::	*[110,86,93, 75,82,65, 87,105,95]*	b:: a:: q::	*[350,360,295]* *[340,225,440]* *[790,920,830]*
g::	*[150,165,138]*	f::	*[190,210,225]*		

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 1394.45 Kc

x(1,3)=1	*s(1)=1*	*y(1)=1*	*z(1,2)=1*	*z(2,1)=1*
x(2,1)=1	*s(2)=1*	*y(2)=1*	*z(1,3)=0.159091*	*z(2,3)=0.840909*
x(2,2)=1		*y(3)=1*		

4. Modely vybraných typů lokačních úloh s více než jedním typem přepravované komodity

V logistické praxi se často vyskytují úlohy lokačního typu, ve kterých je nutno důkladně plánovat rozmístění obslužných středisek v systémech za účelem optimalizace přepravy více druhu komodit.

Tyto typy úloh jsou v odborné literatuře nazývány např. Multi-commodity Uncapacitated Facility Location, Multi-commodity Capacitated Single-echelon Facility Location, Multi-objective Facility Location Problem, Warehouse Location Problem, Multiproduct Two-stage Distribution-Location Problem. Z uvedené množiny úloh budou v předložené bakalářské práci prezentovány tři úlohy označovaná v literatuře jako Multi-commodity Uncapacitated Facility Location (dvě varianty), Multiproduct Two-stage Distribution-Location Problem.

4.1 Multi-commodity Uncapacitated Facility Location (varianta I – MUF)

V MUF je definována dopravní síť, která obsahuje množinu zákazníků a množinu lokalit vhodných pro umístění zdrojů, které mohou vyrábět více komodit. To znamená, že MUF je zobecněním SUFL. Cílem úlohy je zvolit nejvhodnější lokality pro rozmístění zdrojů v síti a přiřadit zákazníky k těm zdrojům tak, aby hodnota celkových nákladů na provoz zdrojů a na zásobování zákazníků různými komoditami byla minimální. V dané úloze se vychází z toho, že pokud bude zdroj v provozu, bude produkovat pouze jednu komoditu.

Zdroj: AIKENS, C [3]

Symboly v modelu mají následující významy:

I – množina lokalit vhodných pro umístění zdrojů

J – množina zákazníků

K – množina komodit

n – počet zákazníků

c_{ijk} – celkové náklady na uspokojení požadavku zákazníka $j \in J$, co se týče komodity $k \in K$ zdrojem umístěným v lokalitě $i \in I$

f_{ik} – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$ za účelem produkce komodity $k \in K$

x_{ijk} – podíl požadavku zákazníka $j \in J$, co se týče komodity $k \in K$, který je uspokojen zdrojem umístěným v lokalitě $i \in I$

y_{ik} – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$ za účelem produkce komodity $k \in K$, pokud je po ukončení optimalizačního výpočtu $y_{ik} = 1$, potom zdroj v lokalitě $i \in I$

bude produkovat komoditu $k \in K$, pokud je $y_{ik} = 0$, potom zdroj v lokalitě $i \in I$ komoditu $k \in K$ produkovat nebude

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y) \;=\; \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} c_{ijk}x_{ijk} \;+\sum_{i\in I}\sum_{k\in K} f_{ik}y_{ik} \qquad (86)$$

za podmínek:

$$\sum_{i\in I} x_{ijk} = 1 \qquad \text{pro } j \in J, k \in K \qquad (87)$$

$$\sum_{k\in K} y_{ik} \le 1 \qquad \text{pro } i \in I \qquad (88)$$

$$\sum_{j\in J} x_{ijk} \le n \cdot y_{ik} \qquad \text{pro } i \in I, k \in K \qquad (89)$$

$$x_{ijk} \;\ge 0 \qquad \text{pro } i \in I, j \in J, k \in K \qquad (90)$$

$$y_{ik} \in \{0,1\} \qquad \text{pro } i \in I, k \in K \qquad (91)$$

Funkce (86) je účelovou funkcí modelu a je složena ze dvou částí. První část reprezentuje náklady spojené se zásobováním zákazníků. Druhá část reprezentuje náklady spojené se zprovozněním zdrojů v lokalitách pro výrobu komodit. Skupina podmínek (87) zabezpečuje, že celkový požadavek zákazníka na jednotlivé komodity bude uspokojen. Počet podmínek (87) se rovná součinu počtu zákazníků a komodit (požaduje-li každý zákazník každou komoditu). Skupina podmínek (88) zajišťuje, že každý zdroji bude produkovat maximálně jeden druh komodity. Počet podmínek (88) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupina podmínek (89) je tvořena vazebními podmínkami. Když bude zdroji umístěnému v lokalitě $i \in I$ nařízeno produkovat komoditu $k \in K$ pro uspokojení požadavku zákazníka $j \in J$, potom budou vynaloženy náklady na jeho provozování za účelem produkce komodity $k \in K$. A naopak, když nebudou vynaloženy náklady na provozování zdroje $i \in I$ za účelem produkce komodity $k \in K$, potom z daného zdroje nebude komoditou $k \in K$ zásobován žádný zákazník. Počet podmínek (89) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a komodit (je-li zdroj v každé lokalitě schopen produkovat každou komoditu). Skupiny podmínek (90) a (91) vymezují definiční obory proměných použitých v modelu. Počet podmínek (90) se rovná součinu počtu lokalit vhodných pro umístění zdrojů, počtu komodit a počtu zákazníků, počet podmínek (91) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a komodit.

Celkový počet omezujících podmínek v modelu je $|I|$ + 2$|I||K|$ + $|J||K|$ + $|I||J||K|$.

Text programu pro typ lokační úlohy MUF v programovacím jazyce Mosel má následující tvar:

model MUF
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..2
komodita=1..2
c:array(zdroj,zakaznik,komodita)of real

```
x:array(zdroj,zakaznik,komodita)of mpvar
f:array(zdroj,komodita)of real
y:array(zdroj,komodita)of mpvar
n:real
end-declarations
forall(j in zakaznik, k in komodita) sum(i in zdroj)x(i,j,k)=1
forall(i in zdroj)sum(k in komodita)y(i,k)<=1
forall(k in komodita,i in zdroj)sum(j in zakaznik)x(i,j,k)<=n*y(i,k)
forall(k in komodita,i in zdroj, j in zakaznik)x(i,j,k)>=0
forall(k in komodita,i in zdroj)y(i,k)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik,k in komodita)c(i,j,k)*x(i,j,k)+sum(i in zdroj,k in komodita)f(i,k)*y(i,k)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik, k in komodita)writeln("x(",i,",",j,",",k,")=",getsol(x(i,j,k)))
forall(i in zdroj, k in komodita)writeln("y(",i,",",k,")=",getsol(y(i,k)))
end-model
```

Pro hodnoty konstant:

```
c::  [95,110,125,80,        f::  [150,165,        n:=  2
      125,130,94,81,              134,172,
      86,94,135,140]              105,158]
```

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 681 Kc

x(1,1,2)=1   x(1,2,2)=1   x(3,1,1)=1   x(3,2,1)=1   y(1,2)=1   y(3,1)=1

4.2 Multi-commodity Uncapacitated Facility Location (varianta I – MUFLP)

Úloha MUFLP je zobecněním úlohy MUF. Základní rozdíl spočívá v tom, že v tomto případě jeden zdroj může produkovat dvě nebo více komodit současně. Tyto komodity v každém zdroji vyvolávají různé fixní náklady na svou výrobu, kromě toho ještě v úloze figurují fixní náklady na zprovoznění zdrojů. Cílem úlohy je rozmístit zdroje na výrobu komodit a přiřadit zákazníky k těm zdrojům tak, aby hodnota celkových nákladů na provoz zprovozněných zdrojů a na zásobování zákazníků různými komoditami byla minimální.

Zdroj: KLOSE, A. a A. DREXL [10]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů

J – množina zákazníků

K – množina komodit

c_{ijk} – celkové náklady na uspokojení požadavku zákazníka $j \in J$ na komoditu $k \in K$ zdrojem umístěným v lokalitě $i \in I$

q_{ki} – fixní náklady zdroje v lokalitě $i \in I$ na produkci komodity $k \in K$

f_i – fixní náklady na zprovoznění zdroje v lokalitě $i \in I$

x_{ijk} – podíl požadavku zákazníka $j \in J$ na komoditu $k \in K$, který je upokojen zdrojem v lokalitě $i \in I$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

z_{ki} – bivalentní proměnná reprezentující poskytování komodity $k \in K$ zdrojem v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $z_{ki} = 1$, potom zdroj v lokalitě i $\in I$ poskytuje komoditu $k \in K$ pokud je $z_{ki} = 0$, potom zdroj v lokalitě $i \in I$ neposkytuje komoditu $k \in K$

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y,z) = \sum_{i\in I}\sum_{j\in J}\sum_{k\in K} c_{ijk}x_{ijk} + \sum_{i\in I}\sum_{k\in K} q_{ki}z_{ki} + \sum_{i\in I} f_i y_i \qquad (92)$$

za podmínek

$$\sum_{i\in I} x_{ijk} = 1 \qquad \text{pro } j \in J, k \in K \qquad (93)$$

$$z_{ki} \le y_i \qquad \text{pro } i \in I, k \in K \qquad (94)$$

$$x_{ijk} \le z_{ki} \qquad \text{pro } i \in I, j \in J, k \in K \qquad (95)$$

$$x_{ijk} \ge 0 \qquad \text{pro } i \in I, j \in J, k \in K \qquad (96)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (97)$$

$$z_{ki} \in \{0,1\} \qquad \text{pro } i \in I, k \in K \qquad (98)$$

Funkce (92) je účelovou funkcí modelu a je složena ze tří částí. První část reprezentuje náklady spojené se zásobováním zákazníků, druhá část reprezentuje fixní náklady na produkci jednotlivých komodit ve zprovozněných zdrojích a třetí část reprezentuje náklady spojené se zprovozněním zdrojů v lokalitách. Skupina podmínek (93) zabezpečuje, že požadavek zákazníka na jednotlivé komodity bude uspokojen. Počet podmínek (93) se rovná součinu počtů zákazníků a komodit. Skupina podmínek (94) zabranuje tomu, aby komodity byly poskytovány ze zdrojů, které nejsou zprovozněny. A naopak, bude-li některý ze zákazníků zásobován ze zdroje minimálně jednou komoditou, musí být daný zdroj v provozu. Počet podmínek (94) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a komodit. Skupina podmínek (95) zajišuje, že bude-li požadavek zákazníka na určitou komoditu plněn ze zdroje, bude daný zdroj produkovat danou komoditu. A naopak, když daný zdroj nebude produkovat danou komoditu, nebude také danou komoditou zásobován daný zákazník. Počet podmínek (95) se rovná součinu počtů lokalit vhodných pro umístění zdrojů, zákazníků a komodit. Skupiny podmínek (96), (97) a (98) zavádějí definiční obory proměnných. Počet podmínek (96) je roven součinu počtů lokalit vhodných pro umístění zdrojů, počtu komodit a počtu zákazníků, počet podmínek (97) se rovná počtu lokalit vhodných pro umístění zdrojů a počet podmínek (98) se rovná součinu počtu lokalit a komodit.

Celkový počet omezujících podmínek v modelu je $|I| + 2|I||K| + |J||K| + 2|I||J||K|$.

Text programu pro typ lokační úlohy MUFLP v programovacím jazyce Mosel má následující tvar:

```
model MUFLP
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..2
komodita=1..2
c:array(zdroj,zakaznik,komodita)of real
x:array(zdroj,zakaznik,komodita)of mpvar
q:array(komodita,zdroj)of real
z:array(komodita,zdroj)of mpvar
f:array(zdroj)of real
y:array(zdroj)of mpvar
end-declarations
forall(k in komodita,j in zakaznik)sum(i in zdroj)x(i,j,k)=1
forall(k in komodita,i in zdroj)z(k,i)<=y(i)
forall(k in komodita,j in zakaznik,i in zdroj)x(i,j,k)<=z(k,i)
forall(k in komodita,j in zakaznik,i in zdroj)x(i,j,k)>=0
forall(i in zdroj)y(i)is_binary
forall(k in komodita,i in zdroj)z(k,i)is_binary
celkove_naklady:=sum(k in komodita,j in zakaznik,i in zdroj)x(i,j,k)*c(i,j,k)+sum(i in zdroj,k in komodita)q(k,i)*z(k,i)+sum(i in zdroj)f(i)*y(i)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik, k in komodita)writeln("x(",i,",",j,",",k,")=",getsol(x(i,j,k)))
forall(i in zdroj, k in komodita)writeln("z(",k,",",i,")=",getsol(z(k,i)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

c::	*[150,210,165,310, 220,135,145,190, 305,180,205,270]*	*q::*	*[110,95, 65,125, 80,75]*	*f::*	*[220,195,235]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 1060 Kc

x(2,1,1)=1	*x(2,2,1)=1*	*z(1,2)=1*	*y(2)=1*
x(2,1,2)=1	*x(2,2,2)=1*	*z(2,2)=1*	

4.3 Multiproduct Two-stage Distribution-Location Problem (MTSDLP)

Úloha MTSDLP je kombinací dvoustupňové multikomoditiní dopravní úlohy a lokační úlohy s tím, že rozmísťována jsou distribuční centra v místech ležících na možných trasách mezi zdroji a zákazníky.

Cílem úlohy je zvolit nejvhodnější rozmístění distribučních center a přiřadit zákazníky k distribučním centrům a zdrojům tak, aby celkové náklady na zásobování zákazníka a na provoz distribučních center byly minimální.

Zdroj: HINDI K. S. a T. BASTA [8]

Symboly v modelu mají následující významy:

I – množina zdrojů

J – množina zákazníků

L – množina potenciálních lokalit vhodných pro umístění distribučních center

K – množina komodit

c_{ilk} – jednotkové přepravní náklady komodity $k \in K$ ze zdroje $i \in I$ do distribučního centra v lokalitě $l \in L$

t_{jkl} – jednotkové přepravní náklady komodity $k \in K$ z distribučního centra v lokalitě $l \in L$ zákazníkovi $j \in J$

b_{jk} – požadavek zákazníka $j \in J$ na komoditu $k \in K$

f_l – fixní náklady na zprovoznění distribučního centra v lokalitě $l \in L$

g_l – jednotkové provozní náklady na komoditu v distribučním centru v lokalitě $l \in L$

p_l – kapacita distribučního centra v lokalitě $l \in L$

a_{ik} – kapacita zdroje $i \in I$ týkající se komodity $k \in K$

x_{ilk} – proměnná modelující množství (počet jednotek) komodity $k \in K$ přepravené ze zdroje $i \in I$ do distribučního centra v lokalitě $l \in L$

v_{jkl} – proměnná modelující množství (počet jednotek) komodity $k \in K$ přepravené z distribučního centra v lokalitě $l \in L$ k zákazníkovi $j \in J$

y_l – bivalentní proměnná reprezentující provozování distribučního centra v lokalitě $l \in L$, pokud je po ukončení optimalizačního výpočtu $y_l = 1$, potom je distribuční centrum v lokalitě $l \in L$ v provozu, pokud je $y_l = 0$, potom distribuční centrum v lokalitě $l \in L$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\ f(x,y,v) = \sum_{i\in I}\sum_{l\in L}\sum_{k\in K} c_{ilk}x_{ilk} + \sum_{j\in J}\sum_{k\in K}\sum_{l\in L} t_{jkl}v_{jkl} + \sum_{i\in I}\sum_{l\in L}\sum_{k\in K} g_l x_{ilk} + \sum_{l\in L} f_l y_l \quad (99)$$

za podmínek

$$\sum_{i\in I}\sum_{k\in K} x_{ilk} \le y_l p_l \qquad \text{pro } l \in L \qquad (100)$$

$$\sum_{l\in L} x_{ilk} \le a_{ik} \qquad \text{pro } i \in I, k \in K \qquad (101)$$

$$\sum_{i\in I} x_{ilk} = \sum_{j\in J} v_{jkl} \qquad \text{pro } k \in K, l \in L \qquad (102)$$

$$\sum_{l\in L} v_{jkl} = b_{jk} \qquad \text{pro } j \in J, k \in K \qquad (103)$$

$$x_{ilk} \ge 0 \qquad \text{pro } i \in I, k \in K, l \in L \qquad (104)$$

$$v_{jkl} \ge 0 \qquad \text{pro } j \in J, k \in K, l \in L \qquad (105)$$

$$y_l \in \{0,1\} \qquad \text{pro } l \in L \qquad (106)$$

Funkce (99) je účelovou funkcí modelu a je složená ze čtyř částí. První část reprezentuje celkové náklady spojené s přepravou komodit ze zdrojů do distribučních center. Druhá část reprezentuje celkové náklady na zásobování zákazníků komoditami z distribučních center. Třetí část reprezentuje celkové náklady na zpracování toků komodit v distribučních centrech. Čtvrtá část reprezentuje celkové fixní náklady na zprovoznění distribučních center. Skupina podmínek (100) zajistí, že pokud je distribuční centrum zprovozněno, tak tok komodit přes dané distribuční centrum nebude překročen. Počet podmínek (100) se rovná počtu lokalit vhodných pro umístění distribučních center. Skupina podmínek (101) zabezpečuje, že toky komodit ze zdrojů nepřekročí jejich kapacity na dané komodity. Počet podmínek (101) se rovná součinu počtů komodit a zdrojů. Skupina podmínek (102) zabezpečuje kontinuitu toků komodit v distribučních centrech. Počet podmínek (102) se rovná součinu počtů distribučních center a komodit. Skupina podmínek (103) zajišťuje, že požadavky zákazníků na jednotlivé komodity budou splněny. Počet podmínek (103) se rovná součinu počtů zákazníků a komodit.

Skupina podmínek (104), (105) a (106) vymezují definiční obory proměnných použitých v modelu. Počet podmínek (104) se rovná součinu počtů lokalit vhodných pro umístění distribučních center, komodit a zdrojů. Počet podmínek (105) se rovná součinu počtů lokalit vhodných pro umístění distribučních center, zákazníků a komodit. Počet podmínek (106) se rovná počtu lokalit vhodných pro umístění distribučních center.

Celkový počet omezujících podmínek v modelu je $2|L| + |I||K| + |J||K| + |K||L| + |I||K||L| + |J||K||L|$.

Text programu pro typ lokační úlohy MTSDLP v programovacím jazyce Mosel má následující tvar:

```
model MTSDLP
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..2
komodita=1..2
centrum=1..2
c:array(zdroj,centrum,komodita)of real
x:array(zdroj,centrum,komodita)of mpvar
```

```
t:array(zakaznik, komodita,centrum)of real
v:array(zakaznik, komodita,centrum)of mpvar
g:array(centrum)of real
f:array(centrum)of real
y:array(centrum)of mpvar
p:array(centrum)of real
a:array(zdroj,komodita)of real
b:array(zakaznik,komodita)of real
end-declarations
forall(l in centrum)sum(i in zdroj,k in komodita)x(i,l,k)<=y(l)*p(l)
forall(i in zdroj,k in komodita)(sum(l in centrum)x(i,l,k)<=a(i,k))
forall(k in komodita,l in centrum)sum(i in zdroj)x(i,l,k)=sum(j in zakaznik)v(j,k,l)
forall(k in komodita,j in zakaznik)sum(l in centrum)v(j,k,l)=b(j,k)
forall(k in komodita,j in zakaznik, l in centrum)v(j,k,l)>=0
forall(k in komodita,j in zakaznik, l in centrum)x(j,k,l)>=0
forall(l in centrum)y(l)is_binary
celkove_naklady:=sum(i in zdroj,k in komodita, l in centrum)c(i,l,k)*x(i,l,k) +sum(j in zakaznik, k in komodita, l in centrum) t(j,k,l)*v(j,k,l)+ sum(i in zdroj,k in komodita, l in centrum)g(l)*x(i,l,k)+sum(l in centrum) f(l)*y(l)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,k in komodita,l in centrum)writeln("x(",i,",",l,",",k,")=",getsol(x(i,l,k)))
forall(j in zakaznik,k in komodita,l in centrum)writeln("v(",j,",",k,",",l,")=",getsol(v(j,k,l)))
forall(l in centrum)writeln("y(",l,")=",getsol(y(l)))
end-model
```

Pro hodnoty konstant:

c::	*[15,20,25,30, 18,23,48,16, 36,24,11,10]*	t::	*[11,13,18,22, 25,18,16,25]*	a::	*[160,180, 220,130, 90,160]*	p::	*[280,310]*
g::	*[250,190]*	b::	*[80,95, 105,80]*	f::	*[310,450]*		

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 83785 Kc

x(1,1,1)=50	*x(3,2,1)=90*	*v(1,1,1)=50*	*v(2,1,2)=105*	*y(1)=1*
x(1,2,1)=45	*x(3,2,2)=160*	*v(1,1,2)=30*	*v(2,2,2)=80*	*y(2)=1*
x(2,2,2)=15		*v(1,2,2)=95*		

5. Lokační úlohy s více optimalizačními kritérii

V logistické praxi se často vyskytují úlohy lokačního typu, které mají za cíl hledat optimální řešení z pohledu více různorodých kritérií.

Tyto typy úloh jsou v odborné literatuře nazývány např. Multi-objective Solid Transportation Problem, Biobjective Obnoxious Facility Location Problem, Multi-objective Emergency Location-transportation Problem, *k*-Balanced Center Location Problem. Z uvedené množiny úloh bude v předložené bakalářské práci prezentována úloha označovaná v literatuře jako Biobjective Obnoxious Facility Location Problem.

Biobjective Obnoxious Facility Location Problem (BOOFLP)

Model BOOFLP byl vytvořen pro řešení úloh spojených s odstraněním odpadu, např. nemocničního. Jedná se tedy o reverzní úlohu k běžné lokační úloze zabávající se zpravidla distribucí zboží. Předpokladem úlohy, je existence sítě vrcholů, kde každý vrchol produkuje určité množsví odpadu, který je třeba pravidelně přepravovat do cross-docking center jejichž počet je předem znám, ale není známa jejich pozice, je však k dispozici množina vrcholů, kde by daná centra mohla být rozmístěna. Platí, že každý zákazník může být přiřazen pouze jednomu cross-docking centru. Cílem úlohy je navrhnout nejvhodnější vrchol (y) pro umístění cross-docking center v síti a přiřadit jim zákazníky tak, aby negativní důsledky ze sváženého zboží působily na co nejmenší populaci a přitom náklady na přepravu byly minimální.

Zdroj: MEDAGLIA, A. L., J. G. VILLEGAS a D. M. RODRÍGUEZ-COCA [14]

Symboly v modelu mají následující významy:

I – množina vrcholů produkujícících odpad

J – množina potenciálních vrcholů vhodných pro umístění cross-docking centra

d_i – množství odpadu vytvořeného vrcholem $i \in I$

a_j – kapacita cross-docking centra ve vrcholu $j \in J$

s_j – populace cross-docking centra ve vrcholu $j \in J$

p – počet cross-docking center, které je třeba rozmístit v síti

c_{ij} – přepravní náklady mezi vrcholem $i \in I$ a cross-docking centrem ve vrcholu $j \in J$

y_j – bivalentní proměnná reprezentující provozování cross-docking centra ve vrcholu $j \in J$, pokud je po ukončení optimalizačního výpočtu $y_j = 1$, potom je cross-docking centrum ve vrcholu $j \in J$ v provozu, pokud je $y_j = 0$, potom cross-docking centrum ve vrcholu $j \in J$ v provozu není

x_{ij} – bivalentní proměnná reprezentující přiřazení vrcholu $i \in I$ k cross-docking centru ve vrcholu $j \in J$, pokud je po ukončení optimalizačního výpočtu $x_{ij} = 1$, potom je vrchol $i \in I$ přiřazen k cross-docking centru ve vrcholu $j \in J$, pokud je $x_{ij} = 0$, potom vrchol $i \in I$ není přiřazen k cross-docking centru ve vrcholu $j \in J$

Matematicky lze model úlohy zapsat ve tvaru:

$$min\ f_1(x,y) = \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \tag{107}$$

$$min\ f_2(x,y) = \sum_{j \in J} s_j y_j \tag{108}$$

za podmínek

$$\sum_{j \in J} y_j = p \tag{109}$$

$$\sum_{j \in J} x_{ij} = 1 \quad \text{pro } i \in I \tag{110}$$

$$\sum_{i \in I} d_i x_{ij} \leq a_j y_j \quad \text{pro } j \in J \tag{111}$$

$$y_j \in \{0,1\} \quad \text{pro } i \in I \tag{112}$$

$$x_{ij} \in \{0,1\} \quad \text{pro } j \in J, i \in I \tag{113}$$

Funkce (107) a (108) jsou účelové funkce modelu. První účelová funkce reprezentuje náklady spojené se svozem odpadu. Druhá účelová funkce reprezentuje velikost populace, která je ovlivněna cross-docking centry. Skupina podmínek (109) zajistí, že počet nově zprovozněných cross-docking center se bude rovnat předem danému číslu. V modelu je obsažena vždy pouze jedna podmínka (109). Skupina podmínek (110) zabezpečuje, že každý vrchol bude přiřazen jen k jednomu cross-docking centru. Počet podmínek (110) se rovná počtu vrcholů. Skupina podmínek (111) zabezpečuje, že množství odpadu přepraveného cross-docking centru nepřekročí kapacitu cross-docking centra a zároveň zajistí, že když cross-docking centrum nebude v provozu, potom danému vrcholu nebude přiřazen žádný vrchol produkující odpad. Počet podmínek (111) se rovná počtu vrcholů vhodných k umístění cross-docking centra. Skupiny podmínek (112) a (113) vymezují definiční obory proměnných. Počet podmínek (112) se rovná počtu vrcholů vhodných k umístění cross-docking centra. Počet podmínek (113) se rovná součinu počtů vrcholů a vrcholů vhodných k umístění cross-docking center.

Celkový počet omezujících podmínek v modelu je 1 + 2$|I|$ + $|J|$ + $|I||J|$.

Text programu pro typ lokační úlohy BOOFLP v programovacím jazyce Mosel má následující tvar:

```
model BOOFLP
uses "mmxprs"
declarations
vrchol=1..4
cdc=1..3
c:array(vrchol,cdc)of real
x:array(vrchol,cdc)of mpvar
```

```
s:array(cdc)of real
y:array(cdc)of mpvar
p:real
d:array(vrchol)of real
a:array(cdc)of real
end-declarations
sum(j in cdc)y(j)=p
forall(i in vrchol)sum(j in cdc)x(i,j)=1
forall(j in cdc)sum(i in vrchol)d(i)*x(i,j)=a(j)*y(j)
forall(j in cdc)y(j)is_binary
forall(i in vrchol, j in cdc)x(i,j)is_binary
funkce_jedna:=sum(i in vrchol,j in cdc)c(i,j)*x(i,j)
funkce_dva:=sum(j in cdc)s(j)*y(j)
minimize(funkce_jedna)
minimize(funkce_dva)
writeln("Minimalni naklady jsou: ",getobjval)
forall(i in vrchol,j in cdc)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(j in cdc)writeln("y(",j,")=",getsol(y(j)))
end-model
```

Pro hodnoty konstant:

```
c::  [15,22,34,                    s::  [150,210,165]
      33,16,28,                    a::  [190,170,155]
      19,21,23,                    d::  [85,110,105,60]
      27,14,16]                    p:=  2
```

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Minimalni naklady jsou: 360

x(1,1)=1 *x(2,2)=1* *x(3,1)=1* *x(4,2)=1* *y(1)=1* *y(2)=1*

6. Lokační úlohy speciálního typu

V logistické praxi se často vyskytují úlohy, které se dají velmi obtížně jednoznačně přiřadit do nějaké skupiny lokačních úloh. Například se může jednat o lokační úlohy, které se zabývají spolehlivostí zdrojů, časovými změnami v požadavcích nebo nákladech, dopravními prostředky, rovnoměrným rozdělením zákazníků atd.

Tyto typy úloh jsou v odborné literatuře nazývány např. Uncapacitated Hub Location problem, Risk-averse Facility Location Problem, Dynamic Location Problem, Capacitated P-median Problem, Solid Transportation Problem, Balanced Location Problem, Fixed Charge Transportation Problem, Capacitated Vertex P-center Problem. Z uvedené množiny úloh budou v předložené bakalářské práci prezentovány čtyři úlohy označované v literatuře jako Fixed Charge Transportation Problem, Capacitated Vertex P-center Problem, Capacitated P-median Problem, Uncapacitated Hub Location problem.

6.1 Fixed Charge Transportation Problem (FCTP)

V úloze FCTP je definována dopravní síť, ve které se nachází množina zdrojů (již rozmístěných) a množina zákazníků. Přeprava kladného počtu jednotek po určité trase je podmíněna vynaložením fixních nákladů. Úlohou je naplánovat přiřazení zákazníků zdrojům (trasy) a počty přepravených jednotek na nich tak, aby celkové náklady na zásobování zákazníků a zprovoznění tras byly minimální.

Zdroj: ADLAKHA, V., K. KOWALSKI a B. LEV [2]

Symboly v modelu mají následující významy:

I – množina zdrojů

J – množina zákazníků

a_i – množství komodity ve zdroji $i \in I$

b_j – požadavek zákazníka $j \in J$ na komoditu

c_{ij} – jednotkové přepravní náklady mezi zdrojem $i \in I$ a zákazníkem $j \in J$

o_{ij} – náklady na zprovoznění trasy mezi zdrojem $i \in I$ a zákazníkem $j \in J$

x_{ij} – počet jednotek komodity přepravených zákazníkovi $j \in J$ ze zdroje $i \in I$

y_{ij} – bivalentní proměnná reprezentující existenci toku mezi zdrojem $i \in I$ a zákazníkem $j \in J$, pokud je po ukončení optimalizačního výpočtu $y_{ij} = 1$, potom tok mezi zdrojem $i \in I$ a zákazníkem

$j \in J$ existuje, pokud je po skončení optimalizačního výpočtu $y_{ij} = 0$, potom tok zboží mezi zdrojem $i \in I$ a zákazníkem $j \in J$ neexistuje

Matematicky lze model úlohy ve vybilancovaném tvaru zapsat ve tvaru:

$$min\, f(x,y) = \sum_{i \in I} \sum_{j \in J} (c_{ij} x_{ij} + o_{ij} y_{ij}) \tag{114}$$

za podmínek

$$\sum_{j \in J} x_{ij} = a_i \quad \text{pro } i \in I \tag{115}$$

$$\sum_{i \in I} x_{ij} = b_j \quad \text{pro } j \in J \tag{116}$$

$$x_{ij} \leq min\{a_i, b_j\} y_{ij} \quad \text{pro } i \in I, j \in J \tag{117}$$

$$x_{ij} \geq 0 \quad \text{pro } i \in I, j \in J \tag{118}$$

$$y_{ij} \in \{0,1\} \quad \text{pro } i \in I, j \in J \tag{119}$$

Funkce (114) je účelovou funkcí modelu, která je složena ze dvou částí. První část reprezentuje náklady na zásobování zákazníka. Druhá část reprezentuje náklady spojené se zprovozněním tras mezi zákazníky a zdroji. Skupina podmínek (115) zabezpečuje, že kapacity zdrojů budou vyčerpány. Počet podmínek (115) se rovná počtu zdrojů. Skupina podmínek (116) zabezpečuje, že požadavky zákazníků budou splněny. Počet podmínek (116) se rovná počtu zákazníků. Skupina podmínek (117) vytváří vazební podmínky. Když se mezi zdrojem $i \in I$ a zákazníkem $j \in J$ přepravuje kladný počet jednotek, potom podmínka zajistí, že fixní náklady na zprovoznění trasy budou vynaloženy, když náklady na zprovoznění trasy nebudou vynaloženy, potom nebude možné na trase přepravovat kladný počet jednotek. Výraz $min\{a_i, b_j\}$ reprezentuje maximální počet jednotek, které lze přepravit na dané trase. Pokud by proměnná y_{ij} byla vynásobena menším kladným číslem, než je uvedený výraz, potom by podmínka mohla omezovat nejvýhodnější (např. maximální) počet jednotek přepravovaných v dané relaci. Počet podmínek (117) se rovná součinu počtů zákazníků a zdrojů. Skupiny podmínek (118) a (119) vymezují definiční obory proměnných v modelu. Počet podmínek (118) se rovná součinu počtů zdrojů a zákazníků, počet podmínek (119) se rovná součinu počtů zdrojů a zákazníků.

Celkový počet omezujících podmínek v modelu je $|I|$+ $|J|$ + 3$|I||J|$.

Text programu pro typ lokační úlohy FCTP v programovacím jazyce Mosel má následující tvar:

```
model FCTP
uses "mmxprs"
declarations
zdroj=1..3
zakaznik=1..3
c:array(zdroj,zakaznik)of real
o:array(zdroj,zakaznik)of real
x:array(zdroj,zakaznik)of mpvar
y:array(zdroj,zakaznik)of mpvar
a:array(zdroj)of real
```

```
b:array(zakaznik)of real
end-declarations
forall(i in zdroj)sum(j in zakaznik)x(i,j)=a(i)
forall(j in zakaznik)sum(i in zdroj)x(i,j)=b(j)
forall(i in zdroj, j in zakaznik) do
if a(i)<b(j) then x(i,j)<=a(i)*y(i,j)
else x(i,j)<=b(j)*y(i,j)
end-if
end-do
forall(i in zdroj,j in zakaznik)x(i,j)>=0
forall(i in zdroj, j in zakaznik)y(i,j)is_binary
celkove_naklady:=sum(i in zdroj,j in zakaznik)(c(i,j)*x(i,j)+o(i,j)*y(i,j))
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj,j in zakaznik)writeln("y(",i,",",j,")=",getsol(y(i,j)))
end-model
```

Pro hodnoty konstant:

c::	*[25,17,19, 31,10,26, 18,30,23]*	*o::*	*[80,115,95, 65,70,120, 85,105,100]*
a::	*[800,690,720]*	*b::*	*[840,650,720]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 36775 Kc

x(1,1)=80	*x(1,3)=720*	*x(2,1)=40*	*x(2,2)=650*	*x(3,1)=720*
y(1,1)=1	*y(1,3)=1*	*y(2,1)=1*	*y(2,2)=1*	*y(3,1)=1*

6.2 Capacitated Vertex P-center Problem (CPC)

V úloze CPC je definována dopravní síť, která obsahuje množinu zákazníků a množinu lokalit vhodných pro umístění zdrojů s předem zadaným počtem nově zprovozněných zdrojů. Cílem úlohy CPC je zvolit nejvhodnější lokality pro rozmístění zdrojů a přiřadit zákazníky k těmto rozmístěným zdrojům tak, aby maximální vzdálenost mezi zprovozněným zdrojem a přiděleným zákazníkem byla minimální.

Zdroj: AYKUT ÖZSOY, F. a M. C. PINAR [4]

Symboly v modelu mají následující významy:

I – množina potenciálních lokalit vhodných pro umístění zdrojů

J – množina zákazníků

p – počet zdrojů, které se mají nově umístit v lokalitách

d_{ij} – vzdálenost mezi lokalitou $i \in I$ a zákazníkem $j \in J$

b_j – požadavek zákazníka $j \in J$

a_i – kapacita zdroje v lokalitě $i \in I$

z – maximální vzdálenost mezi zprovozněným zdrojem a k němu přiřazeným zákazníkem

x_{ij} – bivalentní proměnná reprezentující přiřazení zákazníka $j \in J$ ke zdroji zprovoněném v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $x_{ij} = 1$, potom je zákazník $j \in J$ přiřazen ke zdroji v lokalitě $i \in I$, pokud je $x_{ij} = 0$, potom zákazník $j \in J$ není přiřazen ke zdroji v lokalitě $i \in I$

y_i – bivalentní proměnná reprezentující provozování zdroje v lokalitě $i \in I$, pokud je po ukončení optimalizačního výpočtu $y_i = 1$, potom je zdroj v lokalitě $i \in I$ v provozu, pokud je $y_i = 0$, potom zdroj v lokalitě $i \in I$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x, y, z) = z \qquad (120)$$

za podmínek

$$\sum_{i \in I} x_{ij} = 1 \qquad \text{pro } j \in J \qquad (121)$$

$$x_{ij} \le y_i \qquad \text{pro } i \in I, j \in J \qquad (122)$$

$$\sum_{i \in I} y_i \le p \qquad (123)$$

$$\sum_{i \in I} d_{ij} x_{ij} \le z \qquad \text{pro } j \in J \qquad (124)$$

$$\sum_{j \in J} b_j x_{ij} \le a_i \qquad \text{pro } i \in I \qquad (125)$$

$$x_{ij} \in \{0,1\} \qquad \text{pro } i \in I, j \in J \qquad (126)$$

$$y_i \in \{0,1\} \qquad \text{pro } i \in I \qquad (127)$$

$$z \ge 0 \qquad (128)$$

Funkce (120) je účelovou funkcí modelu a reprezentuje maximální vzdáleností mezi zprovozněným zdrojem a k němu přiřazeným zákazníkem. Skupina podmínek (121) zabezpečuje, že každý zákazník bude přiřazen pouze k jednomu zdroji. Počet podmínek (121) se rovná počtu zákazníků. Skupina podmínek (122) zajistí, že když zákazník bude přiřazen lokalitě, potom v dané lokalitě bude zprovozněn zdroj. A dále zajistí, že když nebude v lokalitě zprovozněn zdroj, tak k ní nebude přiřazen žádný zákazník. Počet podmínek (122) se rovná součinu počtů zákazníků a lokalit vhodných pro umístění zdrojů. Skupina podmínek (123) zajistí, že počet nově zprovozněných zdrojů nepřekročí předem stanovenou hodnotu. V modelu bude právě jedna podmínka typu (123). Skupina podmínek (124) vytváří vazbu mezi přiřazením zákazníků a účelovou funkcí. Počet podmínek (124) se rovná počtu zákazníků. Skupina podmínek (125) zajistí, že celkový požadavek zákazníků, který uspokojuje zdroj, nepřekročí jeho kapacitu. Počet podmínek (125) se rovná počtu lokalit vhodných pro umístění zdrojů. Skupiny podmínek (126), (127) a (128) vymezují definiční obory proměných použitých v modelu. Počet podmínek (126) se rovná součinu počtů lokalit vhodných pro umístění zdrojů a zákazníků, počet podmínek (127) se rovná počtu lokalit vhodných pro umístění zdrojů a podmínka (128) je v modelu pouze jedna.

Celkový počet omezujících podmínek v modelu je 2 + 2 $|I|$ + 2$|J|$ + 2$|I||J|$.

Text programu pro typ lokační úlohy CPC v programovacím jazyce Mosel má následující tvar:

```
model CPC
uses "mmxprs"
declarations
zdroj=1..5
zakaznik=1..3
x:array(zdroj,zakaznik)of mpvar
y:array(zdroj)of mpvar
p:real
d:array(zdroj,zakaznik)of real
z:mpvar
b:array(zakaznik)of real
a:array(zdroj)of real
end-declarations
forall(j in zakaznik)sum(i in zdroj)x(i,j)=1
forall(i in zdroj,j in zakaznik)x(i,j)<=y(i)
sum(i in zdroj)y(i)<=p
forall(j in zakaznik)sum(i in zdroj)d(i,j)*x(i,j)<=z
forall(i in zdroj)sum(j in zakaznik)b(j)*x(i,j)<=a(i)
forall(i in zdroj)y(i)is_binary
forall(i in zdroj, j in zakaznik)x(i,j)is_binary
z>=0
minimize(z)
writeln("Maximalni vzdalenost mezi zakaznikem a pridelenem k nemu zdroji je: ",getobjval)
forall(i in zdroj,j in zakaznik)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(i in zdroj)writeln("y(",i,")=",getsol(y(i)))
end-model
```

Pro hodnoty konstant:

d:: *[15,20,18,*
31,16,22,
25,11,35,
30,28,12,
19,26,32]

p:= 4
b:: *[120,95,135]*
a:: *[250,190,290,320,180]*

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Maximalni vzdalenost mezi zakaznikem a pridelenem k nemu zdroji je: 15

x(1,1)=1 *x(3,2)=1* *x(4,3)=1* *y(1)=1* *y(3)=1* *y(4)=1*

6.3 Capacitated P-median Problem (CPMP)

V úloze CPMP je definována dopravní síť a předem daný počet zdrojů, který je třeba v dané síti rozmístit. Cílem úlohy CPMP je rozmístit zadaný počet zdrojů v síti vrcholů tak, aby součet vzdaleností mezi zdroji a vrcholy k nim přiřazeným byl minimální. V úloze CPMP se předpokládá, že všechny zdroje mají stejnou kapacitu.

Zdroj: LORENA, L. A. N. a E. L. F. SENNE [12]

Symboly v modelu mají následující významy:

M – množina všech vrcholů

p – počet zdrojů, které mají být umístěny ve vrcholech

b_i – požadavek vrcholu $i \in M$

a – kapacita každého možného zdroje $j \in M$

d_{ij} – vzdálenost mezi vrcholem $i \in M$ a přiřazeným zdrojem $j \in M$

x_{ij} – bivalentní proměnná reprezentující přiřazení zdroje $j \in M$ k vrcholu $i \in M$, pokud je po ukončení optimalizačního výpočtu $x_{ij} = 1$, potom je zdroj $j \in M$ přiřazen k vrcholu $i \in M$, pokud je $x_{ij} = 0$, potom zdroj $j \in M$ není přiřazen k vrcholu $i \in M$

y_j – bivalentní proměnná reprezentující provozování zdroje ve vrcholu $j \in M$, pokud je po ukončení optimalizačního výpočtu $y_j = 1$, potom je zdroj ve vrcholu $j \in M$ v provozu, pokud je $y_j = 0$, potom zdroj ve vrcholu $j \in M$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y) = \sum_{i \in M} \sum_{j \in M} d_{ij} x_{ij} \tag{129}$$

za podmínek

$$\sum_{j \in M} x_{ij} = 1 \qquad \text{pro } i \in M \tag{130}$$

$$\sum_{i \in M} b_i x_{ij} \le a y_j \qquad \text{pro } j \in M \tag{131}$$

$$\sum_{j \in M} y_j = p \tag{132}$$

$$x_{ij} \in \{0,1\} \qquad \text{pro } i \in M, j \in M \tag{133}$$

$$y_j \in \{0,1\} \qquad \text{pro } j \in M \tag{134}$$

Funkce (129) je účelovou funkcí modelu a vyjadřuje celkovou vzdálenost mezi vrcholy a jim přiřazenými zdroji. Skupina podmínek (130) zabezpečuje, že každý vrchol bude přiřazen pouze jednomu zdroji (zdroj může být umístěn přímo i do obsluhovaného vrcholu). Počet podmínek (130) se rovná počtu vrcholů. Skupina podmínek (131) zajistí, že pokud je zdroj zprovozněný, tak součet požadavků přiřazených vrcholů nepřekročí jeho kapacitu. Počet podmínek (131) se rovná počtu zdrojů. Skupina podmínek (132) zaručí, že počet nově zprovozněných zdrojů se rovná předem stanovené hodnotě. Počet podmínek (132) se rovná jedné. Skupiny podmínek (133) a (134) vymezují definiční obory proměných použitých v modelu. Počet podmínek (133) se rovná součinu počtů zdrojů a vrcholů. Počet podmínek (134) se rovná počtu zdrojů.

Celkový počet omezujících podmínek v modelu je 3$|M|$ + $|M||M|$.

Text programu pro typ lokační úlohy CPMP v programovacím jazyce Mosel má následující tvar:

```
model CPMP
uses "mmxprs"
declarations
vrchol=1..5
d:array(vrchol,vrchol)of real
x:array(vrchol,vrchol)of mpvar
b:array(vrchol)of real
y:array(vrchol)of mpvar
p:real
a:real
end-declarations
forall(i in vrchol)sum(j in vrchol)x(i,j)=1
forall(j in vrchol)sum(i in vrchol)b(i)*x(i,j)<=a*y(j)
sum(j in vrchol)y(j)=p
forall(j in vrchol)y(j)is_binary
forall(j in vrchol,i in vrchol)x(i,j)is_binary
celkova_vzdalenost:=sum(i in vrchol,j in vrchol)d(i,j)*x(i,j)
minimize(celkova_vzdalenost)
writeln("Celkova vzdalenost je : ",getobjval," km")
forall(i in vrchol,j in vrchol)writeln("x(",i,",",j,")=",getsol(x(i,j)))
forall(j in vrchol)writeln("y(",j,")=",getsol(y(j)))
end-model
```

Pro hodnoty konstant:

d::	*[0,18,16,10,9, 22,0,18,31,15, 17,11,0,33,36, 30,13,14,0,26, 29,18,16,35,0]*	b:: a:= p:=	*[15,20,10,35,25]* 100 2

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkova vzdalenost je: 33 km

x(1,5)=1 x(2,2)=1 x(3,2)=1 x(4,2)=1 x(5,5)=1 y(2)=1 y(5)=1

6.4 Uncapacitated Hub Location problem (UHLP)

Model UHLP se používá nejvíce pro distribuční systémy typu hub&spoke. Cílem úlohy je najít nejvhodnější rozmístění hubů v síti a přiřadit zákazníky k těm zdrojům a hubům tak, aby hodnota celkových nákladů na zásobování vrcholů a náklady na zprovoznění hubů byla minimální.

Zdroj: KLOSE, A. a A. DREXL [10]

Symboly v modelu mají následující významy:

M – množina vrcholů

H – množina vrcholů, které mohou být huby $H \subseteq M$

v_{ij} – tok komodity mezi vrcholy $i, j \ \in M$

f_k – fixní náklady na zprovoznění hubu ve vrcholu $k \in H$

c_{ikmj} – jednotkové přepravní náklady z vrcholu $i \in M$ do vrcholu $j \in M$ přes huby ve vrcholech $k, m \in H$, platí, že $c_{ikmj} = c_{ik} + \alpha c_{km} + c_{mj}$, kde α – parametr $0 \leq \alpha \leq 1$

x_{ikmj} – podíl toku komodity z vrcholu $i \in M$ do vrcholu $j \in M$, který je přepravován přes huby ve vrcholech $k, m \in H$

y_k – bivalentní proměnná reprezentující provozování hubu ve vrcholu $k \in H$, pokud je po ukončení optimalizačního výpočtu $y_k = 1$, potom je hub ve vrcholu $k \in H$ v provozu, pokud je $y_k = 0$, potom hub ve vrcholu $k \in H$ v provozu není

Matematicky lze model úlohy zapsat ve tvaru:

$$min\, f(x,y) \ = \sum_{i \in M} \sum_{k \in H} \sum_{m \in H} \sum_{j \in M} v_{ij}\big(c_{ik} + \alpha c_{km} + c_{mj}\big) x_{ikmj} + \sum_{k \in H} f_k y_k \tag{135}$$

za podmínek

$$\sum_{k \in H} \sum_{m \in H} x_{ikmj} = 1 \qquad \text{pro } i \in M, j \in M \tag{136}$$

$$x_{ikmj} \leq y_k \qquad \text{pro } i \in M, j \in M, m \in H, k \in H \tag{137}$$

$$x_{ikmj} \leq y_m \qquad \text{pro } i \in M, j \in M, m \in H, k \in H \tag{138}$$

$$x_{ikmj} \geq 0 \qquad \text{pro } i \in M, j \in M, m \in H, k \in H \tag{139}$$

$$y_k \in \{0,1\} \qquad \text{pro } k \in H \tag{140}$$

Funkce (135) je účelovou funkcí modelu a je složená ze dvou částí. První část reprezentuje náklady spojené se zásobováním vrcholů přes huby. Druhá část jsou náklady spojené se zprovozněním hubů ve vrcholech. Skupina podmínek (136) zabezpečuje, že celkový požadavek vrcholu bude uspokojen. Počet podmínek (136) se rovná čtverci počtu vrcholů. Skupiny podmínek (137) a (138) zajistí, že tok komodit přes hůby nebude existovat, pokud nebudou zřízené tyto huby. Počet podmínek (137) a (138) se rovná součinu čtverci počtu vrcholů a čtverci počtu vrcholů na huby. Skupina podmínek (139) ukazuje, že podíl toku komodit přes huby může nabývat hodnot rovných nebo větších než nula. Počet podmínek (139) se rovná součinu čtverci počtu vrcholů a čtverci počtu vrcholů na huby. Skupina podmínek (140) definuje definiční obor proměnné. Počet podmínek (140) se rovná počtu vrcholů na huby.

Celkový počet počet omezujících podmínek v modelu je $|H|$ + $|M||M|$ + 3 $|M||M||H||H|$

Text programu pro typ lokační úlohy UHLP v programovacím jazyce Mosel má následující tvar:

```
model UHLP
uses "mmxprs"
declarations
vrchol=1..3
hub=1..2
v:array(vrchol,vrchol)of real
x:array(vrchol,hub,hub,vrchol)of mpvar
f:array(hub)of real
y:array(hub)of mpvar
cik:array(vrchol,hub)of real
a=0.8                                        !parametr
ckm:array(hub,hub)of real
cmj:array(hub,vrchol)of real
end-declarations
forall(i in vrchol,j in vrchol)sum(k in hub,m in hub)x(i,k,m,j)=1
forall(i in vrchol,k in hub,m in hub,j in vrchol)x(i,k,m,j)<=y(k)
forall(i in vrchol,k in hub,m in hub,j in vrchol)x(i,k,m,j)<=y(m)
forall(i in vrchol,k in hub,m in hub,j in vrchol)x(i,k,m,j)>=0
forall(k in hub)y(k)is_binary
celkove_naklady:=sum(i in vrchol,j in vrchol,k in hub,m in hub) (cik(i,k)+a*ckm(k,m)+cmj(m,j))
*x(i,k,m,j)*v(i,j) + sum(k in hub)f(k)*y(k)
minimize(celkove_naklady)
writeln("Celkove naklady jsou: ",getobjval," Kc")
forall(i in vrchol,j in vrchol,k in hub,m in hub) writeln("x(",i,",",k,",",m,",",j,")=",getsol (x(i,k,m,j)))
forall(k in hub)writeln("y(",k,")=",getsol(y(k)))
end-model
```

Pro hodnoty konstant:

v::	*[50,65,98, 84,72,36, 75,81,92]*	*f::*	*[150,230]*	cik::	*[15,23, 31,19, 18,24]*	cmj::	*[15,19,21, 18,35,17]*
		ckm::	*[36,42, 25,28]*				

byly dosaženy následující výsledky (v bakalářské práci budou vypsány pouze proměnné, které po skončení optimalizačního výpočtu nenabyly nulových hodnot):

Celkove naklady jsou: 39600.4 Kc

x(1,2,1,1)=1	*x(2,2,1,1)=1*	*x(3,2,1,1)=1*	*y(1)=1*
x(1,2,1,2)=1	*x(2,2,1,2)=1*	*x(3,2,1,2)=1*	*y(2)=1*
x(1,2,2,3)=1	*x(2,2,2,3)=1*	*x(3,2,2,3)=1*	

7. Závěr

Problematika plánování zdrojů v distribučních systémech je uznána jako jedna z klíčových oblastí spojených s návrhem efektivních logistických řetězců. Investice do nových zdrojů, příp. přemístění stávajících zdrojů je velmi nákladný a citlivý proces. V průběhu projektování musí být identifikována správná místa určená k umístění zdrojů vzhledem k polohám zákazníků, každému zdroji musí být stanovena kapacita na základě tendence vývoje požadavků a musí mu být přidělena často značná část firemního kapitálu, která následně neohrozí chod firmy. S ohledem na danou okolnost i byla vytvořena tato bakalářská práce s cílem poskoutnout stručný přehled základních lokačních úloh s jejich detailním popisem a s praktickou ukázkou textů programů v programovacím jazyce Mosel.

Základní lokační úlohy na dopravní síti jsou úlohy, které mají za účel rozmístit určitý počet zdrojů s omezenou nebo neomeznou kapacitou na jedné úrovni tak, aby byly splněny požadavky zákazníků na homogenní komoditu. Požadavek zákazníka může být, v závislosti na typu úlohy, uspokojen jedním nebo více zdroji současně. Zásobování zákazníků komoditou vyvolává náklady na přepravu, stejně jako i provozování zdrojů v lokalitách vyvolává fixní náklady. Lokační úlohy daného typu se snaží minimalizovat celkovou hodnotu vzniklých nákladů při dodržení všech omezujících podmínek.

Skutečnému logistickému řetězci však spíše odpovídají lokační úlohy, které se zabývají rozmístěním zdrojů ve více úrovních. Nejčastěji se jedná o výrovní závody na vyšších úrovních a o sklady nebo depa na nižších úrovních, přes které pak bude probíhat proces zásobování zákazníků. V dané situaci přepravní náklady vznikají jak mezi zdroji vyšší a nižší úrovně, tak i mezi zdroji nižší úrovně a zákazníky spolu s fixními náklady na zprovoznění zdrojů na obou úrovních. Cílem úloh je, analogicky jako u základních typů lokačních úloh, minimalizace celkových nákladů vzniklých při zásobování zákazníků.

Výpočetně složitějšími jsou lokační úlohy s heterogenními komoditami. To znamená, že zdroje na jedné nebo více úrovních produkují více než jednu komoditu za účelem uspokojení požadavků zákazníků. Vzhledem k typu úlohy zdroj může produkovat jednu nebo více komodit, což pak vyvolává dodatečné fixní náklady na každou další komoditu.

V lokačních úlohách se nejčastěji nachází pouze jedna účelová funkce, ale není to pravidlem. Existuje řada lokačních úloh obsahujících více účelových funkcí. Současně s minimalizací nákladů může být minimalizován negativní efekt na obyvatelstvo (zejména u přepravy odpadů) nebo přepravní doba.

Poslední v práci obsaženou skupinou lokačních úloh tvoří úlohy se speciálními účelovými funkcemi. Tyto účelové funkce místo obvyklé minimalizace nákladů mají za účel např. minimalizaci maximální vzdálenosti od zdroje k zákazníkovi, minimalizaci celkové vzdálenosti mezi zákazníky a zdroji nebo minimalizaci nákladů na zprovoznění přepravních tras.

Pro každý zvolený typ lokační úlohy byl v bakalářské práci vytvořen text programu v programovacím jazyce Mosel, se kterým pracuje optimalizační software Xpress-IVE, v němž byly vzorové úlohy s modelovými daty řešeny. (v bakalářské práci jsou vypisovány pouze proměnné, které po skončení optimalizačního výpočtu nabyly nenulových hodnot).

Vzhledem k širokému uplatnění lokačních úloh lze očekávat, že tato bakalářská práce bude přínosná pro odbornou veřejnost, která ji může použít nejen pro detailnější seznámení s problematikou lokačních úloh, ale také pro svou další práci.

Literatura

1) AARDAL, K., Y. POCHET a L.A. WOLSEY. *Capacitated Facility Location: Valid Inequalities and Facets*. INFORMS: *Mathematics of Operations Research,* 1995, **20** (3), 562-582. ISSN 0364-765X. DOI 10.1287/moor.20.3.562.
2) ADLAKHA, V., K. KOWALSKI a B. LEV. *A branching method for the fixed charge transportation problem*. Elsevier Ltd: *Omega,* 2010 **38** (5), 393-397. ISSN 0305-0483. DOI 10.1016/j.omega.2009.10.005.
3) AIKENS, C. *Facility location models for distribution planning*. North-Holland: *European Journal of Operational Research*, 1985, **22**(3), 263-279. ISSN 0377-2217.
4) AYKUT ÖZSOY, F. a M.C. PINAR. *An exact algorithm for the capacitated vertex p -center problem*. Elsevier Ltd: *Computers and Operations Research,* 2006. **33** (5), 1420-1436. ISSN 0305-0548. DOI 10.1016/j.cor.2004.09.035.
5) BARBATI, M. *Models and Algorithms for Facility Location Problems with Equity Considerations.* University of Naples Federico II, School in Industrial Engineering, Department of Industrial Engineering, Program in Science and Technology Management XXV Cycle.
6) BRANDEAU, M.L. a S.S. CHIU. *An Overview of Representative Problems in Location Research*. INFORMS: *Management Science,* 1989, **35** (6), 645-674. ISSN 0025-1909. DOI 10.1287/mnsc.35.6.645.
7) DARBY-DOWMAN K. a H.S. LEWIS. *Lagrangian Relaxation and the Single-Source Capacitated Facility-Location Problem.* Palgrave Macmillan Journals: *Operational Research Society,* 1988, **39** (11), 1035-1040. ISSN 0160-5682. DOI 10.2307/2583202.
8) HINDI K. S. a T. BASTA. *Computationally Efficient Solution of a Multiproduct, Two-Stage Distribution-Location Problem.* Palgrave Macmillan Journals : *The Journal of Operational Research Society.* 1994, **45** (11), 1316-1323. ISSN 0160-5682. DOI 10.2307/2583859.
9) KLOSE, A. *A Lagrangean relax-and-cut approach for the two-stage capacitated facility location problem*. Elsevier B.V: *European Journal of Operational Research,* 2000, **126** (2), 408-421. ISSN 0377-2217. DOI 10.1016/S0377-2217(99)00300-8.
10) KLOSE, A. a A. DREXL. *Facility location models for distribution system design*. Elsevier B.V: *European Journal of Operational Research,* 2005, **162** (1), 4-29. ISSN 0377-2217. DOI 10.1016/j.ejor.2003.10.031.
11) KRATICA, J., D. DUGOŠIJA a A. SAVIĆ. *A new mixed integer linear programming model for the multi level uncapacitated facility location problem*. Elsevier Inc : *Applied Mathematical Modelling,* 2014, **38** (7-8), 2118-2129. ISSN 0307-904X. DOI 10.1016/j.apm.2013.10.012.
12) LORENA, L.A.N. a E.L.F. SENNE. *A column generation approach to capacitated p-median problems*. Elsevier Ltd: *Computers and Operations Research,* 2004, **31** (6), 863-876. ISSN 0305-0548. DOI 10.1016/S0305-0548(03)00039-X.
13) MARÍN, A. a B. PELEGRÍN. *Applying Lagrangian relaxation to the resolution of two-stage location problems*. Kluwer Academic Publishers: *Annals of Operations Research,* 1999, **86** (0), 179-198. ISSN 0254-5330. DOI 10.1023/A:1018998500803.
14) MEDAGLIA, A. L., J. G. VILLEGAS a D. M. RODRÍGUEZ-COCA. *Hybrid biobjective evolutionary algorithms for the design of a hospital waste management network*. Springer US: *Journal of Heuristics,* 2009, **15** (2), 153-176. ISSN 1381-1231. DOI 10.1007/s10732-008-9070-6.
15) OWEN, S.H. a M.S. DASKIN. *Strategic facility location: A review*. Elsevier B.V: *European Journal of Operational Research*, 1998, **111** (3), 423-447. ISSN 0377-2217. DOI 10.1016/S0377-2217(98)00186-6.
16) TEICHMANN, D., A. GROSSO a M. IVAN. *Modely pro řešení rozhodovacích úloh v logistice I. Acta logistica Moravica*, 2011, **1** (2), 56-68. ISSN 1804-8315.
17) UNO, T. a H. KATAGIRI. *Single- and multi-objective defensive location problems on a network*. Elsevier B.V: *European Journal of Operational Research,* 2008, **188** (1), 76-84. ISSN 0377-2217. DOI 10.1016/j.ejor.2007.04.003.

18) VAN ROY, T.J. *A CROSS Decomposition Algorithm for Capacitated Facility Location*. INFORMS: *Operations Research,* 1986, **34** (1), 145-163. ISSN 0030-364X. DOI 10.1287/opre.34.1.145.
19) Xpress-IVE. *FICO* [software]. [přístup 16. února 2019]. 1983. Dostupné z: https://www.fico.com
20) Xpress-IVE*. In: Wikipedia: the free encyclopedia [online]. St. Petersburg (Florida): Wikipedia Foundation, 11. 12. 2006, last edited on 25. 3. 2019* [cit. 2019-02-16]. Dostupné z: https://en.wikipedia.org/wiki/FICO_Xpress