



**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**  
**FAKULTA DOPRAVNÍ**

Bc. Vojtěch Pazdera  
**OPTIMALIZACE LOGISTIKY FIRMY STO, S.R.O.**

Diplomová práce

**2016**



**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

**Fakulta dopravní  
d ě k a n**

Konviktská 20, 110 00 Praha 1

**K617 ..... Ústav logistiky a managementu dopravy**

**ZADÁNÍ DIPLOMOVÉ PRÁCE**  
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení studenta (včetně titulů):

**Bc. Vojtěch Pazdera**

Kód studijního programu a studijní obor studenta:

**N 3710 – LO – Logistika, technologie a management dopravy**

Název tématu (česky): **Optimalizace logistiky firmy Sto, s.r.o.**

Název tématu (anglicky): Optimization of Logistics in Sto, s.r.o.

**Zásady pro vypracování**

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

- Firma Sto, s.r.o., činnost firmy, logistika produktů
- Analýza SWOT firmy Sto, s.r.o., motivace, problematika závozu Moravy a Slezska
- Teoretické možnosti řešení problematiky lokace skladů
- Návrh řešení logistiky firmy Sto, s.r.o.
- Ekonomické zhodnocení návrhu, doporučené řešení

Rozsah grafických prací: podle charakteru tématu diplomové práce

Rozsah průvodní zprávy: minimálně 55 stran textu (včetně obrázků, grafů a tabulek, které jsou součástí průvodní zprávy)

Seznam odborné literatury: Cenek, P., Klima, V., Janáček, J. Optimalizace dopravních a spojových procesů. Univerzita Žilina, 1994  
Volek, J., Linda, B. Teorie grafů: Aplikace v dopravě a veřejné správě. Pardubice, 2012  
Jurová, M. Podniková logistika. Brno, 1993

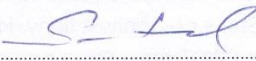
Vedoucí diplomové práce: **doc. Ing. Josef Volek, CSc.**

Datum zadání diplomové práce: **30. června 2015**  
(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 diplomové práce: **1. června 2016**  
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

  
.....  
doc. Ing. Lukáš Týfa, Ph.D.  
vedoucí  
Ústavu logistiky a managementu dopravy



  
.....  
prof. Dr. Ing. Miroslav Svítek, dr. h. c.  
děkan fakulty

Potvrzuji převzetí zadání diplomové práce.

  
.....  
Bc. Vojtěch Pazdera  
jméno a podpis studenta

V Praze dne .....30. června 2015

## **Poděkování**

Na tomto místě bych rád poděkoval vedoucímu diplomové práce doc. Ing. Josefu Volkovi, CSc. za odborné vedení, za veškeré rady, připomínky a trpělivost. Nejvíce oceňuji to, že si vždy našel čas na konzultaci dané problematiky.

Dále bych rád poděkoval řediteli IT a logistiky společnosti Sto, s.r.o. Luboši Kolínskému, za poskytnutí dat a informací a v neposlední řadě zaměstnancům prodejny a logistiky Michalovi Půtovi, Karolíně Matějkové a Miladě Novotné, za konzultaci řešeného problému a za důkladné analyzování současného stavu logistických procesů společnosti Sto, s.r.o.

## **Prohlášení**

Předkládám tímto k posouzení a obhajobě diplomovou práci, zpracovanou na závěr studia na ČVUT v Praze, Fakultě dopravní.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

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).



V Praze, dne 1. 6. 2016

Bc. Vojtěch Pazdera

## **Abstrakt**

Jméno a příjmení autora:	Bc. Vojtěch Pazdera
Název diplomové práce:	Optimalizace logistiky firmy Sto, s.r.o.
Pracoviště:	Fakulta dopravní, České vysoké učení technické v Praze
Vedoucí diplomové práce:	doc. Ing. Josef Volek, CSc.
Rok obhajoby diplomové práce:	2016
Rozsah stran:	91

Téma diplomové práce je optimalizace logistického řetězce společnosti Sto, s.r.o. Analýzou SWOT bylo určeno slabé místo logistických procesů. Identifikovaný problém byl analyzován, modelován a optimalizován pomocí disciplín operačního výzkumu, zejména lokační analýzy, teorie grafů, vícekriteriálního rozhodování a teorie zásob. U každé použité disciplíny jsou vysvětleny základní pojmy a principy tak, aby bylo možné zvládnout jednotlivé fáze výpočtu popsané ve čtvrté kapitole. Za hlavní část diplomové práce lze považovat popis 2. fáze, lokační analýzy. Cílem práce je ekonomické zhodnocení navrhovaných řešení a jejich srovnání se současným stavem.

## **Klíčová slova**

Operační výzkum, lokační analýza, teorie grafů, teorie zásob, vícekriteriální rozhodování, iterativní algoritmus, logistika, optimalizace, logistický řetězec, heuristika.

## **Abstract**

Name and surname of the author:	Bc. Vojtěch Pazdera
Title of the master's thesis:	Optimization of logistics in Sto, s.r.o.
University name and location:	Czech Technical University in Prague, Faculty of Transportation Sciences
Head of the master's thesis:	doc. Ing. Josef Volek, CSc.
Number of pages:	91

The topic of this master's thesis is optimization of logistics in Sto, s.r.o. By means of a SWOT analysis the weak spot of the logistics processes was determined. The identified problem was analyzed, modeled and optimized with the help of the disciplines of operational research, such as location analysis, graph theory, decision theory and inventory theory. For each discipline used the basic concepts and principles were explained, so that it would be possible to manage all phases of the calculation described in the fourth chapter. The main section of the master's thesis is the description of the second phase, i.e. location analysis. The aim of the thesis is the economic evaluation of the proposed solutions and their comparison with the current state.

## **Keywords**

Operational research, location analysis, graph theory, inventory theory, decision theory, iterative algorithm, logistics, optimization, supply chain, heuristics.

## Obsah

1. Úvod .....	8
2. Společnost Sto AG .....	9
2.1 Sto ve světě .....	9
2.2 Cíle společnosti .....	10
2.3 Sto, s.r.o. ....	10
2.4 Prodané množství materiálu .....	11
2.5 Nerovnoměrnost poptávky .....	12
2.6 Doprava.....	12
2.6.1 Outsourcing .....	12
2.6.2 Dopravci externího logistického řešení.....	13
2.6.3 Náklady na dopravu .....	14
2.7 Analýza SWOT.....	15
2.8 Problém dopravy na Moravu a do Slezska a návrh řešení.....	17
3. Teoretická část – Operační výzkum .....	19
3.1 Matematické programování .....	20
3.2 Teorie grafů .....	21
3.2.1 Floydův algoritmus.....	23
3.2.2 Lokační analýza .....	24
3.2.3 Okružní jízdy (Vehicle Routing Problem - VRP) .....	32
3.3 Vícekriteriální rozhodování .....	36
3.3.1 Maximalizační a minimalizační typ kritérií.....	38
3.3.2 Normování hodnot kritérií .....	39
3.3.4 Metody stanovení vah kritérií .....	39
3.3.5 Metody vícekriteriálního hodnocení variant .....	41
3.4 Teorie zásob.....	42
3.4.1 Základní pojmy teorie zásob .....	43
3.4.2 Deterministický a stochastický model zásob .....	44
4. Praktická část – Fáze výpočtu navrhovaných řešení .....	50
0. fáze .....	50
1. fáze – vícekriteriální rozhodování .....	51
2. fáze – lokační analýza .....	54
3. fáze – okružní jízdy s časovými okny (VRPTW).....	61
4. fáze – stochastický model zásob .....	64
5. fáze – ekonomické zhodnocení .....	66
5. Doporučené řešení .....	69

5.1 Výpočet nákladů doporučeného řešení.....	69
5.2 Shrnutí doporučeného řešení .....	70
6. Závěr .....	72
7. Použité zdroje .....	74
7.1 Literatura .....	74
7.2 Internetové zdroje.....	74
8. Seznam obrázků.....	75
9. Seznam grafů .....	75
10. Seznam tabulek .....	75
11. Seznam příloh.....	76
12. Přílohy .....	77



## Seznam použitých zkratek

ADR	Accord Dangereuses Route
AG	Aktiengesellschaft
CVRP	Capacitated Vehicle Routing Problem
EPD	Environmental Product Declarations
EU	Střední hodnota deficitu
IA	Iterated Algorithm
KG	Kommanditgesellschaft
LA	Lokační analýza
LM	Lexikografická metoda
LNS	Large Neighborhood Search
LP	Lineární programování
MGK	Metoda globálního kritéria
s.r.o.	Společnost s ručením omezeným
SA	Simulated Annealing
TSP	Travelling Salesman Problem
VHV	Vícekriteriální hodnocení variant
VKR	Vícekriteriální rozhodování
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows

## 1. Úvod

Logistika, která je jako pojem známa již od řeckých filosofů a naplno začala být využívána v souvislosti s vojenstvím již v 9. století<sup>1</sup>, představuje významnou část firemních procesů, neboť se týká všech komponent oběhového procesu, tzn. dopravy, řízení zásob, manipulace s materiálem, balení, distribuce a skladování. Stručněji lze říci, že se logistika zabývá pohybem zboží a materiálu z místa vzniku na místo spotřeby a s tím souvisejícím informačním tokem.

V dnešní době vysoké konkurence na trhu mezi jednotlivými firmami je velmi důležité optimalizovat firemní procesy. Současný trend rozvoje logistiky říká, že vyhrává jen ten nejrychlejší, nejlevnější a nejefektivnější z pohledu logistické produktivity. Firmy hledají náklady tam, kde jsou nejvíce vidět, hledají možnosti zlepšení, sledují trendy a inspirují se nejlepšími. Největší ztráty v toku zakázky jsou ve většině firem logistické, stejně tak jako náklady na logistiku tvoří často více než 30 % celkových nákladů na produkt. Za hlavní cíle můžeme považovat zvýšení produktivity a snížení nákladů podniku. V moderním pojetí logistiky, respektive její optimalizaci, se stále více firem zaměřuje na procesy skladování a řízení skladového hospodářství a na zásoby se pohlíží jako na jeden z nejdůležitějších aktivních prvků ve schopnosti podniku flexibilně reagovat.

V diplomové práci, psané v rámci projektu Optimalizační úlohy na logistickém řetězci, byly k vyřešení zadané úlohy použity čtyři disciplíny operačního výzkumu. Za hlavní disciplínu lze považovat lokační analýzu, které bylo věnováno nejvíce pozornosti. Dále byla použita teorie grafů, která patří v praxi k často využívaným odvětvím operačního výzkumu. V diplomové práci byla řešena úloha okružních jízd. Další použité disciplíny operačního výzkumu byly vícekriteriální rozhodování a teorie zásob.

Cíl diplomové práce je analyzovat logistické procesy společnosti Sto, s.r.o., určit slabá místa a jejich následná optimalizace. Se společností byla v průběhu projektu navázána úzká spolupráce. K vyřešení problému bylo zapotřebí naprogramování iterativního algoritmu řešícího lokační analýzu. Program lze použít na různé zadání úloh, neboť je napsán tak, aby byl schopen řešit obecné úlohy lokační analýzy v reálném čase. Závěr diplomové práce je navržením nového řešení, které vede k logistickým úsporám společnosti Sto, s.r.o.

---

<sup>1</sup> Císař Leontos VI. (886 – 911).

## **2. Společnost Sto AG**

Stavět zodpovědně, tak zní heslo německé společnosti Sto AG<sup>2</sup>, považující se za předního inovátora a jednoho z mezinárodně nejvýznamnějších výrobců produktů a systémů pro povrchové úpravy budov. Společnost prodává fasádní zateplovací systémy, produkty pro fasádní a interiérové povrchové úpravy, laky a lazury.

Historie společnosti začala ve městě Stühlingen v jižním Německu kolem roku 1954, kdy zakladatel Fritz Stotmeister vytvořil společnost Ispo-Putz KG, vzešlou z cementárny a vápenky, zakoupené jeho otcem Wilhelmem Stotmeisterem roku 1936. Společnost má dnes hlavní sídlo stále ve městě Stühlingen a dodnes jí vlastní členové rodiny Stotmeisterových, což přináší řadu výhod, především v udržování osobních kontaktů s obchodními partnery.

Rok 1962 je považován za rok vzniku značky Sto. Název byl vytvořen z prvních tří písmen příjmení zakladatele Fritze a potažmo celé této německé rodiny. V roce 1970 společnost poprvé začala vyvážet produkty do zahraničí, v roce 1972 se slavnostně otevřela první dceřiná společnost ve Švýcarsku a v roce 1979 v USA, Francii, Rakousku a skandinávských zemích. V roce 2000 pronikla firma Sto AG do Asie založením první pobočky na tomto kontinentě.

### **2.1 Sto ve světě**

Firma Sto AG má v dnešní době zastoupení po celém světě, přičemž největší množství prodejních center se nachází v Německu. Firma zde zaměstnává 211 obchodních zástupců, 30 projektových manažerů, 10 technických prodejců, 38 aplikačních techniků a 12 technických poradců.

Dceřiné společnosti firmy Sto se vyskytují ve všech evropských zemích. V Severní Americe se nacházejí dvě a ve Střední a Jižní Americe sídlí jedna dceřiná společnost a čtyři obchodní partneři. V Asii a v Oceánii společnost Sto zavedla pět dceřiných společností a nachází se zde šestnáct obchodních partnerů.

Výrobní závody byly zřízeny v Německu, Francii, Rakousku, Polsku, Švédsku, Španělsku, Rusku, České republice, USA, Čile, Číně a v Malajsii.

---

<sup>2</sup> AG – akciová společnost v německém jazyce.

## 2.2 Cíle společnosti

Produkty firmy Sto prospívají lidem i životnímu prostředí. A to již od doby, kdy „ochrana životního prostředí“ nebyla zařazena jako pojem ve slovníku. Už v roce 1964 odborníky ve firmě Sto napadlo využít materiál Styropor k zateplování, a od té doby se díky fasádním zateplovacím systémům Sto ušetřilo 150 milionů tun CO<sub>2</sub> [21].

Společnost každoročně investuje nemalé částky do modernizace výrobních hal, což přispívá k energetické efektivitě a úspoře. Firma disponuje odborem životního prostředí, které dohlíží na to, aby výrobní procesy neohrožovaly životní prostředí. Sto dostala jako jedna z prvních firem pro svou minerální omítku označení EPD<sup>3</sup>.

Heslo společnosti zmíněné na začátku druhé kapitoly, stavět zodpovědně, lze vyjádřit tak, že výrobky a systémy mají splňovat požadavky na energetickou efektivitu. Prevenční technologie uvádějí do souladu ekologii a ekonomiku. Mezi hlavní cíle společnosti patří udržení se na prvním místě v technologiích pro povrchové úpravy budov [21].

## 2.3 Sto, s.r.o.

Sto, s.r.o. je dceřiná společnost německé firmy Sto AG působící na území České republiky od roku 2002.

Hlavní sídlo a sklad byly zřízeny v malé obci Dobřejovice ve Středočeském kraji. Ke dni 1. 1. 2010 zde žilo pouhých 880 obyvatel, ale výstavbou nových rodinných domů se v posledních letech tento počet navýšil. Místo hlavního sídla a skladu je strategicky zvolené, neboť lokace na okraji Prahy zaručuje nižší poplatky za pronájem skladových a administrativních prostor a zároveň je dostupnost hlavního města a celé České republiky dobrá. Sídlo a sklad se nachází u dálnice D1 a nájezdu na městský okruh spojující dálnici D1 na exitu 10 s dálnicí D5 na exitu 1. Sklad se nachází společně s administrativními prostorami v moderní budově. Velikost skladu činí 17 010 m<sup>2</sup>. Ve skladu je možné uložit až 854<sup>4</sup> palet.

Právní forma společnosti je s.r.o. Dle webu Evropské databanky [18] zaměstnává společnost 26 – 100 osob a Sto, s.r.o. dosahuje ročně obrátu nad 100 000 tis. Kč. Aktuální přesný počet zaměstnanců činí 38 osob. Na základě těchto dat se společnost řadí do tzv. malých podniků, neboť zaměstnává méně než 50 osob a obrat je nižší než 10 miliónů EUR ročně.

---

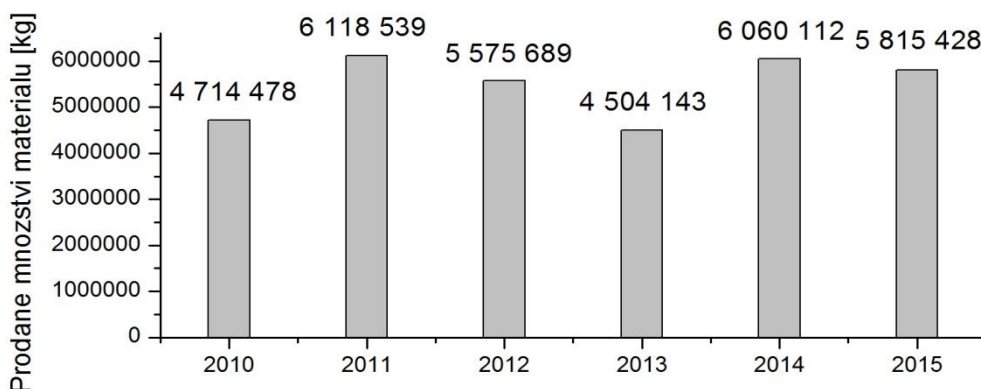
<sup>3</sup> Environmental Product Declarations (Environmentální prohlášení o produktu), vzniká hodnocením LCA (hodnocení životního cyklu výrobku), vydává ho Agentura pro ekologicky šetrné výrobky.

<sup>4</sup> Záleží na uspořádání regálů.

Stejně jako mateřská společnost i tuzemský podnik zákazníkům dodává systémy vnějšího zateplení budov, široký sortiment ušlechtilých omítek a malt, sanační systémy a fasádní barvy. Produkty se do Dobřevovic dováží ze tří hlavních výrobních závodů v Německu. Jedná se o závody ve městě Stühlingen, Donaueschingen a Tollwitz. Výrobní závod Tollwitz se zaměřuje na minerální lepicí a armovací malty, které se vozí na paletách o hmotnosti 900 kg<sup>5</sup>. Ve Stühlingen a Donaueschingen se vyrábí omítky a fasádní barvy. Dodávky se z těchto závodů nakládají na jeden kamion, kde jedna paleta má hmotnost 600 kg.

## 2.4 Prodané množství materiálu

Množství prodaného materiálu má dlouhodobě rostoucí charakter. V následujícím grafu 1 je vidět pokles poptávky v letech 2012 a 2013. Prodané množství se v současné době opět zvedá a v budoucnu se počítá s dalším nárůstem.



Graf 1: Prodané množství zboží v letech 2010-2015 [autor]

Společnost Sto, s.r.o. je závislá na velkých zakázkách, jako jsou opravy a rekonstrukce panelových domů. Snaha firmy je se začít orientovat i na malé zakázky (rekonstrukce a výstavba rodinných domů). Mezi hlavní a nejdůležitější zákazníky společnosti Sto, s.r.o., kteří se podílejí na největším odbytu zboží, patří Skanska a.s., HOLBORN GROUP s.r.o., METROSTAV a.s. a ATRIUM.

Nejprodávanejší materiál je dlouhodobě Levell Duo Plus (1 705 525 kg<sup>6</sup>), následovaný materiálem Levell Duo (1 113 350 kg<sup>7</sup>). Jedná se o malty<sup>8</sup>, které se na stavby prodávají ve velkém množství kvůli větší spotřebě. Malta Levell Duo Plus tvořila v roce 2014 28 % z celkového prodaného množství všech materiálů. Ne vždy je množství prodaného zboží

<sup>5</sup> Na jeden kamion se vejde 26 těchto palet.

<sup>6</sup> Rok 2014

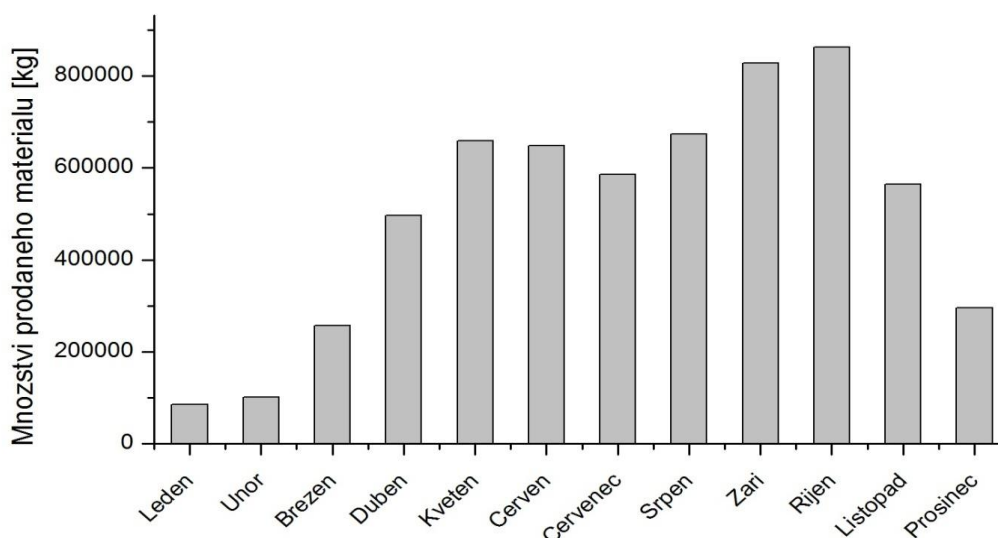
<sup>7</sup> Rok 2014

<sup>8</sup> Slangově lepidlo.

spojeno s obratem firmy; malt se sice prodává největší množství, ale jejich cena za kilogram je oproti jiným produktům nízká. Sto, s.r.o. nabízí velké množství druhů minerálních lepicích a armovacích malt. Tento materiál se ve stavebnictví používá jako spojovací hmota a hmota omítková. Odborněji řečeno se tento materiál využívá pro lepení tepelně izolačních desek na minerální podklady a pro zhotovení tenkovrstvé armovací stěrky<sup>9</sup>. Mezi omítkami jsou dlouhodobě nejprodávanější Stolit a Silcolit.

## 2.5 Nerovnoměrnost poptávky

Jako každá společnost podnikající ve stavebním průmyslu, i Sto, s.r.o. má v průběhu roku nerovnoměrnou poptávku po zboží. Množství poptávaného zboží tvoří křivku typickou pro stavební průmysl. Jedná se o graf se dvěma maximy, která se zpravidla nacházejí v červnu a v říjnu. Obecně lze rok rozdělit na čtyři části, které jsou označeny jako útlum 1 (prosinec - březen), špička 1 (duben - červen), útlum 2 (letní prázdniny) a špička 2 (září - listopad). Nerovnoměrnost poptávky je znázorněna v grafu 2.



Graf 2: Křivka poptávky po materiálu v roce 2014 [autor]

## 2.6 Doprava

### 2.6.1 Outsourcing

Doprava společnosti Sto, s.r.o. je řešená outsourcingem. Jako hlavní důvod preference externího logistického řešení před interním Sto, s.r.o. uvádí důvody nákladové, což bývá nejčastější příčina volby outsourcingu. Jmenovitě jde o náklady na vzdělávání zaměstnanců, náklady spojené s provozováním vlastního vozového parku a náklady spojené s dopravou

<sup>9</sup> Armovací stěrka = vrstva malty na izolantu společně s tkaninou.

nebezpečného zboží podle ADR<sup>10</sup>. Z těchto důvodů je pro společnost Sto, s.r.o. výhodnější dopravu objednávat. Za hlavní výhody outsourcingu lze považovat lepší soustředění se podniku na hlavní činnost, sdílení rizik s dopravcem a uvolnění kapitálových prostředků.

Dopravu stavebního materiálu k zákazníkům nelze jednoduše optimalizovat. Důvod je ten, že stavební společnosti objednávající materiál často mění svá stanoviště. Proto je nezbytné, aby objednávky obsahovaly správné a aktuální adresy, kde daná společnost pracuje a kde čeká dodávku materiálu. Při dodání zboží na špatné místo zbytečně dochází k finančním ztrátám společnosti Sto, s.r.o.

Existují pouze čtyři místa, kam Sto, s.r.o. dodává materiál pravidelně. Jde o dodávky do prodejen s fasádními barvami v Plzni a v Klatovech, dodávky společnosti Haas Fertigbau Chanovice s.r.o. a společnosti ATRIUM. Další problém distribuce zboží stavebním firmám je časté objednávání materiálu na poslední chvíli. Firmy se tak chtějí vyhnout prostojům zaplacených pracovníků na stavbách způsobených špatnou kalkulací potřebného množství stavebního materiálu. Z těchto důvodů je potřeba mít dopravu řešenou spolehlivými dopravci, kteří jsou schopni flexibilně reagovat.

### **2.6.2 Dopravci externího logistického řešení**

Jedním z hlavních využívaných dopravců společností Sto, s.r.o. je Zach Trans, s.r.o. Na trh dopravce přišel zhruba ve stejnou dobu jako Sto, s.r.o. Obě firmy se v České republice rozvíjely společně. I proto panují mezi firmami dobré vztahy a ochota dopravce vždy flexibilně reagovat, spolupracovat a kdykoliv dát k dispozici nákladní vůz ze své flotily. Další výhodou je poskytování přepravy podle ADR, což je vedle ceny a spolehlivosti hlavním kritériem při výběru dopravců. Dalším důležitým dopravcem je JIRSA, TRANS, s.r.o. Tento dopravce začal pro Sto, s.r.o. vozit materiál nedávno. Svoji spolehlivostí se zařadil k prvně jmenovanému dopravci a v současné době mají tyto dva dopravci zakázky rozděleny půl na půl. Zach Trans, s.r.o. a JIRSA TRANS, s.r.o. dohromady pokrývají zhruba 80 % všech zakázek společnosti Sto, s.r.o. a navzájem spolupracují, což viditelně zlepšuje kvalitu služeb. Další využívaní dopravci jsou TOPTRANS EU, a.s., Raben Logistics Czech s.r.o. nebo Doprava na paletách s.r.o.<sup>11</sup> Kamionovou dopravu z Německa na sklad v Dobřejovicích zajišťuje dopravce Trade trans Log s.r.o.

---

<sup>10</sup> Přeprava podlah v kapalném stavu.

<sup>11</sup> Konec spolupráce v roce 2015, kvůli požadavkům, na které Sto, s.r.o. nemohlo přistoupit.

### 2.6.3 Náklady na dopravu

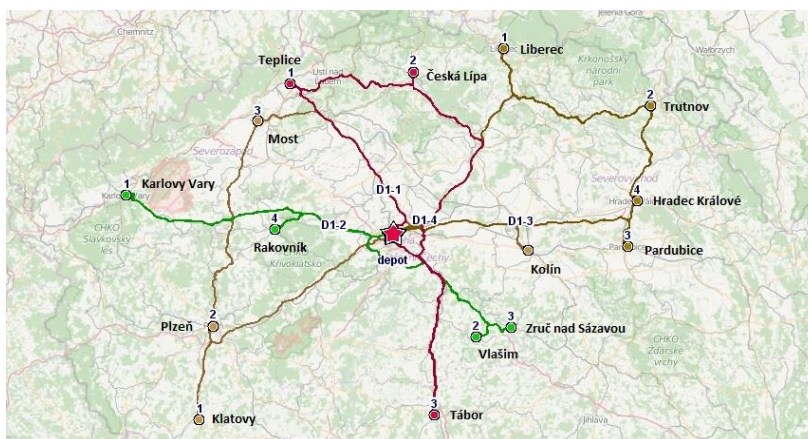
Náklady na dopravu jsou plně hrazeny společností Sto, s.r.o. Obsluha zákazníků je řešena vytvořením tzv. závozů. Ty jsou rozděleny do různých směrů od Dobřejovic tak, aby náklady na dopravu byly co nejmenší a aby byli za týden obslouženi všichni zákazníci v České republice. Místa, s vysokou poptávkou jsou navštěvována minimálně dvakrát týdně. Pořadí vykládky určuje dopravce, podle zvolené trasy. Jedná se o Vehicle Routing Problem (VRP) se stochastickou poptávkou. Firma Sto, s.r.o. určuje hlavní cíle trasy před tím, než je známá skutečná poptávka. Cílem je minimalizovat předpokládanou sumu ujetých kilometrů.

Pracovníci logistického oddělení firmy Sto, s.r.o. přijímají objednávky do různých míst České republiky. Pomocí seznamu směrů závozů sdělí zákazníkovi, jaký den nejdříve bude zdarma zboží dovezeno. Většina stavebních firem již rozdělení závozů zná a snaží se objednávat materiál s časovou rezervou a to nejpozději do 12:00 den před závozem. Jestliže zákazník potřebuje dovézt materiál mimo závoz, hradí dopravu vzniklou navíc ze svých prostředků.

Závozy jsou rozděleny následovně:

- **Pondělí:** Kolín, Hradec Králové, Pardubice, Praha, Morava, Zruč nad Sázavou,
- **úterý:** Karlovy Vary, Most, Teplice, Rakovník, Česká Lípa, Liberec, Tábor,
- **středa:** Plzeň, Klatovy, Kolín, Hradec Králové, Pardubice, Trutnov, Morava,
- **čtvrtek:** Karlovy Vary, Most, Teplice, Rakovník, České Lípa, Liberec, Zruč nad Sázavou, Praha, Vlašim,
- **pátek:** Plzeň, Klatovy, Tábor.

Na obrázku 1 je softwarově vypočítaná úloha okružních jízd pro města uvedená v závozech. Každé město je navštíveno pouze jednou a závoz na Moravu a do Slezska je vynechán. Obrázek slouží pro představu, jak mohou vypadat dopravcem vytvořené trasy při obsluze území Čech. Úloha okružních jízd bude popsána v kapitole 3.2.3.



Obr. 1: Výstup programu Open Door Logistics Studio [autor]



V následujících tabulkách 1 a 2 jsou uvedeny náklady na dopravu k zákazníkům a na sklad po jednotlivých měsících v letech 2013 a 2014.

Tab. 1: Náklady na dopravu k zákazníkům [autor]

	2013	2014
Leden	114 132	97 900
Únor	130 471	107 200
Březen	160 296	215 550
Duben	230 456	340 800
Květen	297 615	472 000
Červen	480 967	592 414
Červenec	491 482	519 392
Srpen	104 959	493 716
Září	739 477	600 434
Říjen	645 863	560 453
Listopad	358 000	465 527
Prosinec	370 775	264 600
<b>Celkem [Kč]</b>	<b>4 124 493</b>	<b>4 729 986</b>

Tab. 2: Náklady na dopravu na sklad [autor]

	2013	2014
Leden	109 128	106 480
Únor	85 477	96 880
Březen	131 780	238 229
Duben	294 922	397 429
Květen	341 030	536 366
Červen	327 513	535 356
Červenec	499 897	395 626
Srpen	515 492	472 396
Září	588 755	594 406
Říjen	570 755	617 746
Listopad	359 575	462 466
Prosinec	169 128	229 001
<b>Celkem [Kč]</b>	<b>3 993 452</b>	<b>4 682 381</b>

Tabulka 3 obsahuje aktuální cenu dopravy za km, v závislosti na nosné hmotnosti použitého automobilu.

Tab. 3: Tabulka cen za km [Sto, s.r.o.]

<b>Tranzit (1,6t)</b>	<b>10 Kč/km</b>
<b>3t</b>	<b>14 Kč/km</b>
<b>6t</b>	<b>16,5 Kč/km</b>
<b>9t</b>	<b>19 Kč/km</b>
<b>Vlek</b>	<b>23 Kč/km</b>
<b>Kamion</b>	<b>25 Kč/km</b>

## 2.7 Analýza SWOT

K analýze firemního prostředí bylo použito analýzy SWOT<sup>12</sup>, která pomáhá zmapovat fungování firmy. Analýza se zaměřuje na silné a slabé stránky společnosti Sto, s.r.o., dále potom na současné příležitosti a hrozby, kterým firma musí čelit. SWOT je komplexní metoda kvalitativního vyhodnocení veškerých relevantních stránek fungování firmy a její současné pozice a slouží k tvorbě strategie [19].

<sup>12</sup> Název je odvozen z počátečních písmen anglických slov strengths, weaknesses, opportunities a threats.

Silné a slabé stránky jsou tzv. interní záležitosti podniku a jsou v přímé kompetenci firmy. Jde například o personální vybavení, marketing, financování podniku, dodavatele, vztah se zákazníky, atd. Naopak do vnějšího prostředí, tedy faktory ležící mimo kontrolu podniku, patří hrozby a příležitosti.

Po konzultaci se zaměstnanci logistiky a ředitelem IT a logistiky, byla provedena analýza SWOT a nalezeny tak problematické oblasti a procesy firmy Sto, s.r.o. Sama společnost vidí své slabé stránky v nákladech navíc spojených s chybně zaslaným materiálem, reklamací či dodávkou na chybnou adresu. Většina zaměstnanců uvedla jako hlavní problém distribuci zboží na Moravu a do Slezska. Protože se jedná o problém s potenciální možností nalezení vysoké úspory, bylo rozhodnuto, že se bude diplomová práce zabývat vytvořením návrhu nového řešení distribuce zboží na Moravu a do Slezska.

Mezi další slabé stránky firmy Sto, s.r.o. patří malá prezentace na internetu, neboť při vyhledávání pomocí nejpoužívanějšího vyhledávače<sup>13</sup> bylo Sto, s.r.o. zobrazeno až po všech svých hlavních konkurentech. Mezi ně patří firmy Weber a Baunit, spol. s r.o.

Přes malou prezentaci na internetu zákazníci společnosti nechybí a naopak stále přibývají. Důvodem je především povědomí stavebních firem o nadstandardní kvalitě<sup>14</sup> výrobků společnosti Sto, s.r.o. Kvalitou se nemohou konkurenti v současné době se společností Sto, s.r.o. srovnávat. Firma Sto, s.r.o. navíc nedávno zavedla na trh levnější verzi omítky Stolit, jejíž cena je rovna cenám nejlepších omítek zmíněných konkurentů, ale je zároveň kvalitnější. Vedle kvality zboží je silnou stránkou i školení zaměstnanců. Obchodní zástupci mají znalosti technických poradců a projektoví manažeři jsou školeni pro komunikaci s architekty. Analýza SWOT firmy Sto, s.r.o. je zapsána v tabulce 4.

Tab. 4: Analýza SWOT firmy Sto, s.r.o.[autor]

Analýza SWOT		
	Silné stránky	Slabé stránky
Vnitřní původ	<ul style="list-style-type: none"> <li>• Kvalita výrobků,</li> <li>• dostatečná technická vybavenost,</li> <li>• pozitivní vnímání značky,</li> <li>• dobré obchodní výsledky,</li> <li>• dobré školení zaměstnanců.</li> <li>• dobré vztahy se zákazníky,</li> <li>• SAP<sup>15</sup>,</li> </ul>	<ul style="list-style-type: none"> <li>• Občasná záměna zásilky či tónu barvy,</li> <li>• nutnost dovážení materiálu z Německa,</li> <li>• <b>závoz Moravy a Slezska,</b></li> <li>• přílišná ochota vyhovět velkým zákazníkům<sup>16</sup>,</li> <li>• malá prezentace na internetu.</li> </ul>

<sup>13</sup> Google.com

<sup>14</sup> Jednodušší práce s materiálem, delší výdrž, omyvatelnost atd.

<sup>15</sup> SAP = podnikový informační systém.

	<b>Příležitosti</b>	<b>Hrozby</b>
<b>Vnější původ</b>	<ul style="list-style-type: none"> <li>• Vzrůstající poptávka po produktech,</li> <li>• vznik nových zákaznických segmentů,</li> <li>• spolupráce s novými dodavateli,</li> <li>• vznik nových distribučních řetězců,</li> <li>• konkurenční boj vytvořením levnějších a zároveň kvalitních produktů.</li> </ul>	<ul style="list-style-type: none"> <li>• Zvyšování cen energií,</li> <li>• recese světové ekonomiky,</li> <li>• vstup nové konkurence na trh,</li> <li>• spojení hráčů na trhu,</li> <li>• cenové války a nečestné jednání konkurence,</li> <li>• ohrožení ze strany dodavatelů,</li> <li>• zvýšení kvality konkurence.</li> </ul>

## 2.8 Problém dopravy na Moravu a do Slezska a návrh řešení

V současném řešení, se Morava a Slezsko zavazí dvakrát týdně a to vždy v pondělí a ve středu. Kvůli velké vzdálenosti skladu v Dobřejovicích a míst dodání jsou náklady na dopravu vysoké. Množství materiálu zasláno v roce 2014 na Moravu a do Slezska tvořilo pouze 15,47 % z celkového objemu a cena za dopravu byla 35,03 %, 1 656 733 Kč (viz tabulka 5).

Tab. 5: Porovnání množství prodeje [autor]

	<b>Celá Česká republika</b>	<b>Morava a Slezsko</b>	<b>[%]</b>
<b>Prodané množství materiálu [kg]</b>	6 060 112	937 749	15,47
<b>Cena dopravy [Kč]</b>	4 729 986	1 656 733	35,03

Doprava na Moravu a do Slezska je realizována následujícími způsoby:

- První způsob je přes spediční firmu. Obsluha je realizována jako úloha okružních jízd se skladem v Dobřejovicích,
- druhou variantou je sloučení více dodávek na jeden kamion. Kamionem (většinou dvanáctitunovým) je zboží přepravováno do Brna. Obsluha je realizována jako úloha okružních jízd se skladem v Brně.

V minulosti byla řešena distribuce zboží na Moravu a do Slezska dvěma sklady Sto, s.r.o. v Brně a v Ostravě. Toto řešení bylo využíváno v období nižší poptávky, sklady se nevyplácelo provozovat a musely být zrušeny. Podle vedení společnosti se nový sklad nevyplatí do té doby, dokud nedosahuje obratu nejméně 20 000 000 Kč<sup>17</sup> ročně. Spotřeba zboží společnosti Sto, s.r.o. má dlouhodobě rostoucí charakter a je proto otázkou, zdali by se v dnešní době nový sklad nevyplatil. Postoj vedení firmy k zavedení nového skladu je skeptický. Jako důvod uvádí pokračující nízkou poptávku na Moravě a ve Slezsku. Zisk

<sup>16</sup> Viz kapitola 2.4. Sto, s.r.o. je závislé na velkých stavbách a stavebních firmách.

<sup>17</sup> Vypočteno z nákladů na provoz skladu (pronájem budovy, platy zaměstnanců, atd.) a přiměřeného zisku.

z prodeje zboží na Moravě a ve Slezsku je v současné době proti nákladům spojeným s provozem skladu (pronájem budovy, platy zaměstnanců) nedostatečný.

Teoretické řešení, které se v tomto případě nabízí, spočívá v pronájmu paletových míst v logistických centrech na Moravě a ve Slezsku. Podmínkou je, aby sklady nabízely mimo uskladnění palet i další služby, jako je například manipulace. Skladování na Moravě a ve Slezsku musí být provozováno externě, tedy tak, aby zde nemuseli být přítomni zaměstnanci Sto, s.r.o. V předchozí kapitole bylo uvedeno, že nejprodávanejší materiály jsou malty. Jsou to navíc materiály, které nevyžadují další úpravy na rozdíl od omítek, které musí zaměstnanci Sto, s.r.o. tónovat. Minerální lepicí a armovací malty se proto jeví jako ideální zboží pro skladování na pronajatých paletových místech a mohou být na sklad dováženy přímo z výrobního závodu v Tollwitz<sup>18</sup>. Tabulka 6 zobrazuje množství prodaných malt v roce 2014 na Moravě a ve Slezsku a náklady na jejich dovoz k zákazníkům.

Tab. 6: Prodané množství malt na Moravě a ve Slezsku v roce 2014 a náklady na dopravu [autor]

Období	Útlum 1	Špička 1	Útlum 2	Špička 2	Celkem
<b>Prodané množství malt [Kg]</b>	46 875	183 950	111 400	145 450	487 675
<b>Náklady na dopravu malt [Kč]</b>	98 157	271 790	202 941	250 267	823 155

V následující kapitole budou popsány a vysvětleny nástroje operačního výzkumu, které byly použity k návrhu nového řešení dopravy na Moravu a do Slezska a k porovnání řešení s řešením, které bylo využíváno v roce 2014.

#### Operační výzkum byl použit pro získání odpovědí na následující otázky:

- V jakém městě nejlépe pronajmout paletová místa? (Lokační analýza a VKR<sup>19</sup>),
- Jaké by byly náklady na distribuci z nově lokalizovaného skladu v roce 2014? (VRP),
- Jaké by byly náklady na skladování v nově lokalizovaném skladě v roce 2014? (Teorie zásob).

<sup>18</sup> Cena Tollwitz – Praha = 8200 Kč, cena Tollwitz – Brno = 14 000 Kč.

<sup>19</sup> Vícekriteriální rozhodování.

### **3. Teoretická část – Operační výzkum**

Operační výzkum (anglický ekvivalent Operational Research, americký ekvivalent Operations Research) je relativně nová vědní disciplína, nebo lépe řečeno soubor vědních disciplín. Hlavní cíl operačního výzkumu lze definovat jako analýzu různých typů rozhodovacích problémů. V odborné literatuře se často pro jednodušší pochopení tohoto pojmu objevuje záměna slov v názvu a operační výzkum se popisuje jako výzkum operací. Operační výzkum se obecně používá pro analýzu a koordinaci prováděných operací v rámci systému. Tyto operace nemohou být prováděny nezávisle (vždy závisí na omezených zdrojích, které jsou operací čerpány a na jiných operacích) [6]. Operační výzkum pomáhá řešiteli nelézt optimální řešení problému, závislého na omezeních a kritériích, které mají přímý vliv na chod systému. Operační výzkum používá nástrojů různých vědních disciplín jako je matematika, statistika, ekonomie a psychologie a kombinuje je s cílem poskytování racionálního základu pro rozhodování při neúplnosti informací.

„Operační výzkum je umění dávat špatné odpovědi na ty otázky, na něž se jinými způsoby získávají odpovědi ještě horší.“ Thomas L. Saaty

Rok vzniku operačního výzkumu není jednoduché určit. První zmínky pocházejí již ze 17. století. Operační výzkum jako pojem byl poprvé použit ve Velké Británii až ve třicátých letech 20. století. Druhá světová válka měla velký vliv na celkovém rozvoji této vědní disciplíny, neboť Velká Británie a USA používali operační výzkum pro řešení složitých logistických, taktických a strategických vojenských problémů a operací. Další důležité období byla padesátá léta minulého století. Operační výzkum se v těchto letech využíval na ekonomickou obnovu válkou zasažených zemí a při poválečném ekonomickém rozvoji. Jako další faktor ovlivňující rozšíření operačního výzkumu se uvádí výrazný rozvoj výpočetní techniky v poválečném období.

#### **Klasifikace disciplín operačního výzkumu**

Operační výzkum používá různých nástrojů a technik a dá se rozdělit do devíti relativně samostatných vědních disciplín. Rozdělení je důsledkem toho, že k řešení problémů je zapotřebí různých specifických přístupů a rozdílného matematického aparátu. Vědci zkoumající operační výzkum věnovali zvláštní důraz matematickému programování, teorii her, vícekritériálnímu rozhodování, teorii hromadné obsluhy (teorie front), teorii grafů, teorii zásob, modelům obsluhy a simulacím. V následujících odstavcích budou některé z těchto disciplín operačního výzkumu popsány.

### 3.1 Matematické programování

Ačkoliv matematické programování nebylo použito k řešení zadaného problému, nelze ho v teoretické části vynechat. Jde o jednu z nejdůležitějších částí operačního výzkumu. Matematické programování řeší optimalizační úlohy hledáním extrému daného kritéria. Matematický model úlohy matematického programování se vždy skládá z kritériální funkce (účelové funkce) a soustavy omezujících podmínek.

Kritériální funkce  $z = f(x)$  je funkce, jejíž extrém (maximum či minimum) je třeba nalézt [6]. Jestliže model obsahuje pouze jednu kritériální funkci, mluvíme o tzv. jednokritériální úloze matematického programování. Kritériální funkce může být buď minimalizační, nebo maximalizační, lineární, nebo nelineární. Jestliže je kritériální funkce lineární, hovoříme o tzv. lineárním programování, jestliže je kritériální funkce nelineární, jedná se o nelineární programování [6].

V praxi se setkáme častěji s lineárním programováním než s programováním nelineárním. Důvodem je jeho snadnější použití a řešení. Mezi nejčastěji se vyskytující úlohy tohoto typu řadíme problém skladby sortimentu, úlohu o vytváření směsi, řezný problém, distribuční úlohy, lokační úlohy a pokrývací úlohy [6]. Dále se vyskytují úlohy týkající se investičních strategií nebo jejich různé kombinace.

Postup sestavování úlohy matematického programování [6]:

A. Analýza problému

B. Sestavení matematického modelu

1. Stanovení proměnných v úloze a jejich jednotky,
2. určení kritéria úlohy,
3. určení omezení úlohy,
4. určení, zdali proměnné použité v úloze podléhají nutnosti splňovat podmínku nezápornosti/celočíselnosti.

Mezi základní pojmy mimo již zmíněné kritériální funkce a příslušných omezujících podmínek patří i přípustné řešení a optimální řešení.

**Přípustné řešení** je takové řešení, které vyhovuje všem podmínkám úlohy, tzn. všem vlastním omezením i podmínkám nezápornosti/celočíselnosti [6].

**Optimální řešení** je přípustné řešení s nejlepší hodnotou účelové funkce (s nejvyšší hodnotou v případě maximalizace a nejnižší hodnotou v případě minimalizace účelové funkce) [6].

Zápis obecného matematického modelu úlohy lineárního programování lze zjednodušit pomocí sumací:

Maximalizovat (minimalizovat):

$$z = \sum_{j=1}^n c_j x_j \quad (1)$$

za podmínek:

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, 2, \dots, m \quad (2)$$

$$x_j \geq 0, \quad j = 1, 2, \dots, n \quad (3)$$

### 3.2 Teorie grafů

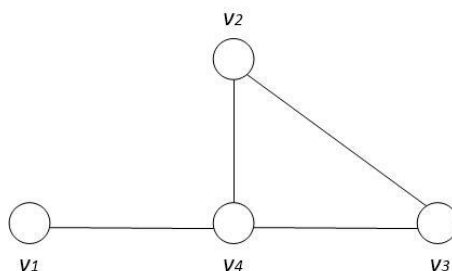
Jak bylo řečeno již v úvodu, teorie grafů je (nejen) v dopravní praxi často používanou disciplínou operačního výzkumu. Historii tohoto vědního oboru začal psát již v 17. století slavný francouzský matematik Pierr de Fermat (1601-1665), který v roce 1629 formuloval lokační problém, který byl vyřešen italským matematikem a fyzikem Evangelistou Torricellim<sup>20</sup> (1608-1647) v roce 1644 definováním tzv. Fermatova-Toricelliho bodu. Další důležité jméno v souvislosti s historií vzniku teorie grafů je Leonhard Euler (1704–1783), který v roce 1736 vyřešil světoznámý problém 7 mostů města Königsburg. Mezi další významné osoby patří R. W. Hamilton (1805-1865). Irský matematik formuloval problém obchodního cestujícího a vyřešil úlohu hlavolamu pravidelného dvanáctistěnu [16].

V úvodu kapitoly definujeme základní pojmy graf, incidence, stupeň vrcholu, délka cesty, vzdálenost dvou vrcholů, dráha a matice vzdáleností, atd.

Konečným **grafem** (dále jen grafem) rozumíme uspořádanou trojici  $G = (V, X, p)$ , kde  $V$  a  $X$  jsou množiny, přičemž  $V$  je konečná neprázdná množina a  $p$  je prosté zobrazení množiny  $X$  do množiny všech neuspořádaných dvojic  $(u, v)$ ,  $u, v \in V$ ,  $u \neq v$ . Prvky množiny  $V$  nazýváme vrcholy grafu  $G$ , prvky množiny  $X$  hranami grafu  $G$  a zobrazení  $p$  incidencí grafu  $G$  [16].

---

<sup>20</sup> Vynálezce barometru.



Obr. 2: Graf [autor]

**Incidence**  $p$  přiřazuje každé hraně grafu neuspořádanou dvojici vrcholů. Platí-li pro incidenci hrany  $h \in X$ ,  $p(h) = (u, v)$ , říkáme, že hrana  $h$  inciduje s vrcholem  $u$  i s vrcholem  $v$  [16].

**Stupněm vrcholu** nazýváme počet hran incidujících s vrcholem  $v \in V$  a označujeme  $st(v)$ . Pro každý graf platí následující vztah mezi počtem hran a stupni vrcholů [16]:

$$\sum_{v_i} st(v_i) = 2q \quad (4)$$

**Délka cesty** mezi dvěma vrcholy  $u, v \in V$  hranově ohodnoceného grafu  $G$  je definována jako součet ohodnocení hran cesty, vyjádřeno analyticky [16]:

$$|m(u, v)| = \sum_{h \in m(u, v)} o(h) \quad (5)$$

**Vzdálenost dvou vrcholů**  $u, v \in V$  v hranově ohodnoceném grafu  $G = (V, X, p)$  je definována:

$$d(u, v) = \min_{m(u, v) \in M} \{ \sum_{h \in m(u, v)} o(h) \}, \quad (6)$$

kde  $M$  je množina všech cest mezi vrcholy  $u$  a  $v$ .

**Nejkratší (minimální) cestou** mezi vrcholy  $u$  a  $v$  v grafu  $G = (V, X, p)$  je cesta  $m^*(u, v) \in M$ , pro kterou platí [16]:

$$\sum_{h \in m^*(u, v)} o(h) = \min_{m(u, v) \in M} \{ \sum_{h \in m(u, v)} o(h) \} \quad (7)$$

Úloha nalezení nejkratší cesty v grafu mezi dvěma vrcholy patří v teorii grafů k základním a nachází uplatnění (nejen) v dopravní praxi (GPS, robotika, počítačová grafika, atd.) Existují dvě základní metody řešení. Prvním a starším algoritmem je Fordova metoda nalezení nejkratší cesty v hranově ohodnoceném grafu z roku 1956. Druhou možností je použití Dijkstrův metody (1959). Dijkstrův algoritmus se obecně mezi programátory považuje za o



něco efektivnější<sup>21</sup> (rychlejší) než Fordův algoritmus. Jako výstup a výsledek obou metod je posloupnost vrcholů a hran hledané nejkratší (minimální) cesty.

**Orientovaný graf** (též orgraf) má podobné matematické vyjádření jako neorientovaný graf, jen množina orientovaných hran se značí místo  $X$  pomocí písmene  $Y$ . Orgrafem rozumíme uspořádanou trojici  $G = (V, Y, p)$ . Orientované hrany oproti neorientovaným umožňují pouze jednosměrný pohyb.

**Dráhu**  $m^*[u, v] \in M$  z vrcholu  $u$  do vrcholu  $v$  orgrafu  $G = (V, Y, p)$  nazveme maximální dráhou, když pro ni platí [16]:

$$|m^*[u, v]| = \sum_{h \in m^*[u, v]} o[h] = \max_{m[u, v] \in M} \{\sum_{h \in m[u, v]} o[h]\}, \quad (8)$$

kde  $M$  je množina všech drah  $m[u, v]$ .

**Matice vzdáleností** (distanční matice) je matice, jejíž prvek  $d_{ij}$  vyjadřuje délku minimální cesty mezi vrcholy  $v_i$  a  $v_j$ , resp. vzdálenost těchto vrcholů [16].

**Matice přímých vzdáleností** je matice grafu  $G$   $D = (d_{ij})_{i,j=1}^n$ , jejíž prvky jsou definovány následovně [16]:

- $d_{ij} = o(h)$ , jestliže  $\exists h \in X$ , pro kterou  $p(h) = (v_i, v_j)$ ,
- $d_{ij} = \infty$ , jestliže  $\nexists h \in X$ , pro kterou  $p(h) = (v_i, v_j)$ ,
- $d_{ij} = 0$ , pro  $i = j$ .

### 3.2.1 Floydův algoritmus

Floyd-Warshallův algoritmus (Floydův algoritmus) slouží k nalezení vzdáleností mezi všemi dvojicemi vrcholů grafu. Jinými slovy tento algoritmus slouží k vytvoření distanční matice grafu  $G$ . Za jeho hlavní výhody se považuje implementační jednoduchost. Autorem algoritmu je slavný americký informatik Robert W. Floyd (1936-2001).

Floydův algoritmus byl v diplomové práci použit k získání distanční matice pro všechny vrcholy obsluhované společností Sto, s.r.o. v roce 2014 na Moravě a ve Slezsku. Program MATLAB, který je jedním z nejlepších programů na řešení matic (jednoduchost a rychlost), obsahuje speciální funkci *distances*. Stačí, aby uživatel ve zdrojovém kódu deklaroval graf

---

<sup>21</sup> Při práci s nezáporným ohodnocením hran. V dopravních úlohách záporné ohodnocení hran neexistuje.

$G$  (parametr funkce), nebo vložil matici přímých vzdáleností. Poté je možné příkazem  $D = \text{distances}(G)$  vytvořit distanční matici  $D$ .

V jiných programovacích jazycích je řešení úlohy složitější a programátor musí znát jednotlivé kroky algoritmu. Ty jsou následovné [21]:

1. Sestavení matice přímých vzdáleností  $C$ , přičemž pro prvky  $c_{ij}$  této matice platí:
  - $c_{ij} = o(h)$  pokud  $i \neq j$ , jestliže  $\exists h \in X$ , pro kterou  $p(h) = (v_i, v_j)$ ,
  - $c_{ij} = \infty$  pokud  $i \neq j$ , jestliže  $\nexists h \in X$ , pro kterou  $p(h) = (v_i, v_j)$ ,
  - $c_{ij} = 0$  pokud  $i = j$ .
2. Zavedeme pomocnou proměnnou  $k$  a položíme  $k = 1$ . Tato proměnná představuje index vrcholu, přes který provádíme přepočty.
3. Provedeme přepočty jednotlivých prvků  $c_{ij}$  matice  $C$  podle pravidla  $c_{ij} = \min\{c_{ij}, c_{ik} + c_{kj}\}$ , přičemž nepočítáme prvky matice, pro které platí  $i = j$  (hlavní diagonála matice) a prvky, pro které platí  $i, j = k$  (leží v řádku či sloupci s indexem  $k$ ), a prvky  $i \neq k$  a  $j \neq k$ , pro které  $c_{ik} = \infty$  a  $c_{kj} = \infty$ .
4. Pokud  $k < n$  ( $n$  je počet vrcholů grafu), potom položíme  $k = k + 1$  a vracíme se zpět ke kroku 3). Je-li  $k = n$ , je výpočet ukončen a poslední získaná matice je hledanou maticí vzdáleností.

### 3.2.2 Lokační analýza

V posledních padesáti letech se lokační analýza stala jednou z nejpoužívanějších a nejvyhledávanějších disciplín operačního výzkumu. Tato disciplína se zabývá rozmístováním (lokací) různých zařízení ve dvou nebo trojrozměrném prostoru.

V lokační analýze se těmto zařízením říká střediska obsluhy nebo také depa. V praxi se může jednat například o požární stanice, nemocnice, školy či sklady. Obecně se umístění dělí na dva případy. První případ je umístění střediska/středisek v geografickém prostoru, ve kterém doposud není/nejsou žádné jiné a druhý případ je umístění depa/dep s respektováním již stávajícího zařízení. Umístění musí vždy vycházet z optimalizace zadaných či zvolených kritérií (optimální vzdálenost, kapacita střediska, výše nákladů apod.) [16].

Mezi základní možné využití lokační analýzy v praxi patří například umístění [16]:

- Výrobních podniků, firem, opraven, servisních středisek,
- skladů a veřejných logistických center,
- nemocnic, zdravotnických středisek, stanic záchranné služby,
- stanic hasičského záchranného sboru, policie,
- čerpacích stanic pohonných hmot, nabíjecích stanic pro elektromobily apod.

Přestože se úlohy lokační analýzy mohou v praxi různě lišit, jejich hlavní cíl zůstává stejný a to umístění jednoho nebo více středisek obsluhy, ze kterého (ze kterých) bude možné obsluhovat vzniklé požadavky. Požadavkem na obsluhu rozumíme například potřebu údržby komunikace v zimních měsících, ale také například vzniklý požár, ohrožení života nebo úraz, potřeba občana výběru hotovosti z bankomatu, atd. Uspokojení těchto požadavků na obsluhu je realizováno ve střediscích obsluhy. Odlišnosti jednotlivých úloh lokační analýzy mohou být především v počtu rozmísťovaných středisek obsluhy, umístění požadavků v geografickém prostoru, kritériu kvality řešení (úloha typu minimax, kde chceme, aby byl čas dojezdu do každého místa minimální) a ve způsobu obsluhy úseků [16].

### **3.2.2.1 Třídy lokačních úloh**

Z předchozích bodů je zapotřebí blíže specifikovat „umístění požadavků v geografickém prostoru.“ V lokační analýze známe tři třídy lokačních modelů v závislosti na použitém prostoru.

První skupina je umísťování střediska obsluhy do kontinuálního prostoru, tedy tak, jak se řeší Fermat-Weberův problém. Řešení těchto lokačních úloh spadá do nelineárního programování.

Druhou skupinu tvoří lokace v sítích, kde není stanovena konečná množina řešení a umístění je možné jak **diskrétní** (ve vrcholech) tak i **spojité** (na hranách grafu). Síť v lokační analýze rozumíme souvislý vrcholově a hranově ohodnocený obyčejný graf  $G = (V, X)$ . Pomocí dopravní sítě, lze vyjádřit běžnou komunikační síť, kde vrcholy reprezentují místa vzniku požadavků na obsluhu (obce, města). Ohodnocení vrcholů  $w(h)$  nejčastěji vyjadřuje počet požadavků na obsluhu za časovou jednotku a ohodnocení hran vyjadřuje délky úseků komunikací.

Třetí třída lokačních úloh je v praxi nejpoužívanější. To je ovlivněno především dostupností pozemků a obecně územními omezeními. Při této lokaci známe konečnou množinu možných

řešení<sup>22</sup> a nazývá se **diskrétní lokace**. Středisko obsluhy se při diskrétní lokaci umísťuje do některého z vrcholů sítě [16].

Abychom se dále mohli podrobněji zabývat úlohami lokační analýzy, je zapotřebí objasnit a definovat základní terminologii disciplíny. Mezi základní pojmy patří mimo středisek obsluhy také atrakční obvod a obsluhované vrcholy a hrany. Vše je přehledně zapsáno a popsáno v následujících odstavcích.

### 3.2.2.2 Základní pojmy lokační analýzy

Mezi základní pojmy lokační analýzy patří **střediska obsluhy** ( $v$ ) neboli depa. Jak bylo již popsáno výše, je tímto pojmem označován objekt, který se lokační analýzou snažíme pomocí algoritmu či modelu umístit na dopravní síť. Jedním z hlavních parametrů úloh je počet a typ těchto umísťovaných objektů. Množina středisek obsluhy se nazývá **množina dep**, kterou označíme  $D_k$ . Počet dep označíme  $k$  a platí pro něj následující vztah:  $1 \leq k \leq p$ , kde  $p = |V|$ .

**Obsluhované vrcholy a hrany** jsou dalším základním pojmem lokačních úloh. Vrcholy lze chápat jako zákazníky, které je třeba zásobovat zbožím, anebo kteří vyžadují dostupnost určité služby. V určitých případech lokační analýzy je zdrojem požadavků celá hrana grafu. Jestliže nastane takovýto případ, hovoříme o hranové lokaci<sup>23</sup> [16].

Dalším důležitým pojmem je **atrakční obvod**. Definice uvádí, že atrakčním obvodem depa  $v \in D_k$  v širším smyslu slova rozumíme množinu vrcholů  $u \in V$  a hran  $h \in X$  sítě, které jsou obsluhovány z depa  $v$ . V užším slova smyslu lze atrakční obvod daného střediska obsluhy, který se značí  $A(v)$ , pokládat za množinu vrcholů  $u \in V$  a hran  $h \in X$  sítě, které jsou obsluhovány pouze z jednoho depa  $v \in D_k$ , pro které platí [16]:

- vrchol  $u \in A(v)$ , pokud neexistuje depo  $w \in D_k$ , pro které  $d(w, u) < d(v, u)$ ,
- hrana  $h \in A(v)$ , pokud neexistuje depo  $w \in D_k$ , pro které  $d(w, h) < d(v, h)$ ,

kde  $d(v, h)$  je vzdálenost hrany  $h$  od depa  $v \in D_k$  a  $d(u, v)$  je vzdálenost vrcholu  $u \in V$  od depa  $v \in D_k$ .

---

<sup>22</sup> Obsahuje optimální řešení úlohy.

<sup>23</sup> Například zimní údržba komunikací.

**Prvotní atrakční obvod**  $A'(v)$  depa  $v \in D_k$  je množina vrcholů  $u \in V$  a hran  $h \in X$  sítě, pro které platí [16]:

- vrchol  $u \in A'(v)$ , pokud neexistuje depo  $w \in D_k$ , pro které  $d(w, u) \leq d(v, u)$ ,
- hrana  $h \in A'(v)$ , pokud neexistuje depo  $w \in D_k$ , pro které  $d(w, h) \leq d(v, h)$ ,

kde  $d(v, h)$  je vzdálenost hrany  $h$  od depa  $v \in D_k$  a  $d(u, v)$  je vzdálenost vrcholu  $u \in V$  od depa  $v \in D_k$ .

**Přidělený atrakční obvod** depa  $v \in D_k$  se značí  $A^*(v)$  je množina vrcholů  $u \in V$  a hran  $h \in X$  sítě, pro které platí [16]:

- $A'(v) \subseteq A^*(v) \subseteq A(v)$  pro každé depo  $v \in D_k$ ,
- $\bigcup_{v \in D_k} A^*(v) = X \cup V$ ,
- $A^*(v) \cap A^*(v) = \emptyset$  pro  $\emptyset; u \neq v, u, v \in D_k$ ,

kde množina  $X$  značí množinu všech hran a množina  $V$  značí množinu všech vrcholů.

V následujících odstavcích budou popsány matematické vztahy a vzorce, které jsou důležité pro stanovování dopravní práce (hodnoty účelové funkce) při výpočtu úloh lokační analýzy.

### 3.2.2.3 Kritérium pro optimalizaci rozmístění dep na síti (Obsluha vrcholů a hran sítě) Obsluha vrcholů sítě

Množinu  $k$  dep  $D_k$  nazveme vrcholově optimálním umístěním  $k$  dep na síti  $G = (V, X)$ , když pro ni platí vztah [16]:

$$f(D_k) = \min_{D'_k} \{f(D'_k)\}, \quad (9)$$

kde  $f(D'_k) = \sum_{v \in D'_k} \sum_{u \in A^*(v)} 2 * d(u, v) * w(u)$ .

### Obsluha hran sítě

Množinu  $k$  dep  $D_k$  nazveme hranově optimálním umístěním  $k$  dep na síti  $G = (V, X)$ , když pro ni platí vztah [16]:

$$g(D_k) = \min_{D'_k} \{g(D'_k)\}, \quad (10)$$

kde  $g(D'_k) = \sum_{v \in D'_k} \sum_{u \in A^*(v)} (2 * d(v, h) + o(h)) * w(h)$ .

K výpočtu úlohy lokační analýzy diplomové práce byl použit první vztah, tedy vztah pro obsluhu vrcholů sítě, neboť lokace byla řešena jako diskrétní. Přesněji řečeno byl tento vztah použit v prosté záměnné heuristice realizované iterativním algoritmem. Metoda bude popsána v kapitole 3.2.2.5.

Vzorec (10) počítá s možnou lokací na hranách sítě a platí zde Hakimiho věta. Hakimiho věta zní [16]:

Nechť pro libovolnou  $k$ -prvkovou množinu bodů (ne vrcholů) sítě –  $Y_k$  jsou funkce  $f(Y_k)$ ,  $g(Y_k)$ , formálně definovány stejně jako  $f(D_k)$  a  $g(D_k)$ . Potom existuje alespoň jedna množina  $k$  vrcholů sítě  $G = (V, X)$ , pro kterou platí:

$$f(D_k) \leq f(Y_k), \text{ resp. } g(D_k) \leq g(Y_k).$$

### 3.2.2.4 Model lokační úlohy jako úloha lineárního programování

Vybraný model lineárního programování popisuje nekapacitní problém lokace středisek obsluhy. V úloze se nezohledňují kapacitní možnosti středisek obsluhy. Model lokační úlohy lze zapsat následovně [16]:

Účelová funkce:

$$\min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + \sum_{i=1}^m f_i y_i \quad (11)$$

Za podmínek:

$$x_{ij} > 0 \quad i = 1, \dots, m; j = 1, \dots, n \quad (12)$$

$$\sum_{i=1}^m x_{ij} = 1 \quad j = 1, \dots, n \quad (13)$$

$$x_{ij} \leq y_i \quad i = 1, \dots, m \quad (14)$$

$$y_i \in \{0,1\} \quad i = 1, \dots, m \quad (15)$$

$$\sum_{j=1}^n d_j x_{ij} \leq s_i y_i \quad i = 1, \dots, m \quad (16)$$

kde:

$J = \{1, 2, \dots, n\}; j = 1, \dots, n$  je množina požadavků na obsluhu

$d_j$  = velikost požadavku  $j$ . zákazníka ( $d_j \geq 0$ ),

$I = \{1, 2, \dots, m\}; i = 1, \dots, m$  je množina lokací středisek obsluhy,

$f_i$ : ( $f_i \geq 0$ ) jsou fixní náklady na zřízení/instalaci střediska obsluhy v místě  $i$ ,

$c_{ij}$  = jednotkové náklady na uspokojení požadavku klienta/požadavku  $j$  zařízením  $i$ ,

$x_{ij}$  = podíl uspokojení klienta/požadavku na obsluhu  $j$  –  $d_j$  střediskem  $i$ ,

$s_i$  = kapacita zařízením/střediska obsluhy v  $i$ ,

$y_i$  = binární proměnná vyjadřující vybudování/instalaci střediska obsluhy v místě  $i$  ( $y_i = 1$ ), v opačném případě ( $y_i = 0$ ).

### 3.2.2.5 Iterativní algoritmus

Prosté heuristické metody (zkráceně heuristiky) řešiteli nezaručují nalezení optimálního řešení a dokonce ani řešení přípustného. Na druhou stranu je bezpochyby jejich předností možnost použití tam, kde je potřeba řešit úlohy velkého rozsahu. Lokační problém je kombinatorickou úlohou třídy  $C_k(p)$  a takovéto úlohy není možné v praxi řešit exaktními metodami. Princip exaktních metod lze vyjádřit jako postupné procházení a vyhodnocování všech možných řešení úlohy, což vede k velice náročnému a dlouhému výpočtu. Například v úloze lokační analýzy, jde o procházení všech kombinací vrcholů a následné počítání příslušných hodnot dopravní práce. Exaktní metody se v odborné literatuře nejčastěji spojují s metodou branch-and-bound<sup>24</sup>.

Jednotlivé heuristické metody se od sebe liší krokem přechodu od jednoho přípustného řešení k jinému a také lokálním kritériem. Mezi nejznámější principy těchto metod patří hladové algoritmy a lokální hledání. Metoda hladového algoritmu spočívá v nalezení optimální lokace prvního depa za pomoci řešitelem zvolené účelové funkce. Další depo se vždy lokalizuje již s umístěným předchozím střediskem obsluhy.

Metody založené na lokálním hledání (local research) jsou o něco sofistikovanější. Využívá se zde hledání přípustného řešení z řešení výchozího (výchozí množina dep). Určení výchozí množiny se provádí jinou lokační metodou, nebo ji volí sám řešitel. Je zde proto nutné, aby měl řešitel základní vědomosti lokační analýzy a věděl, jaké vrcholy grafu přitahují výsledné řešení. Z pravidla jde o vrcholy s největší vahou a vrcholy s největším stupněm, například významné křižovatky.

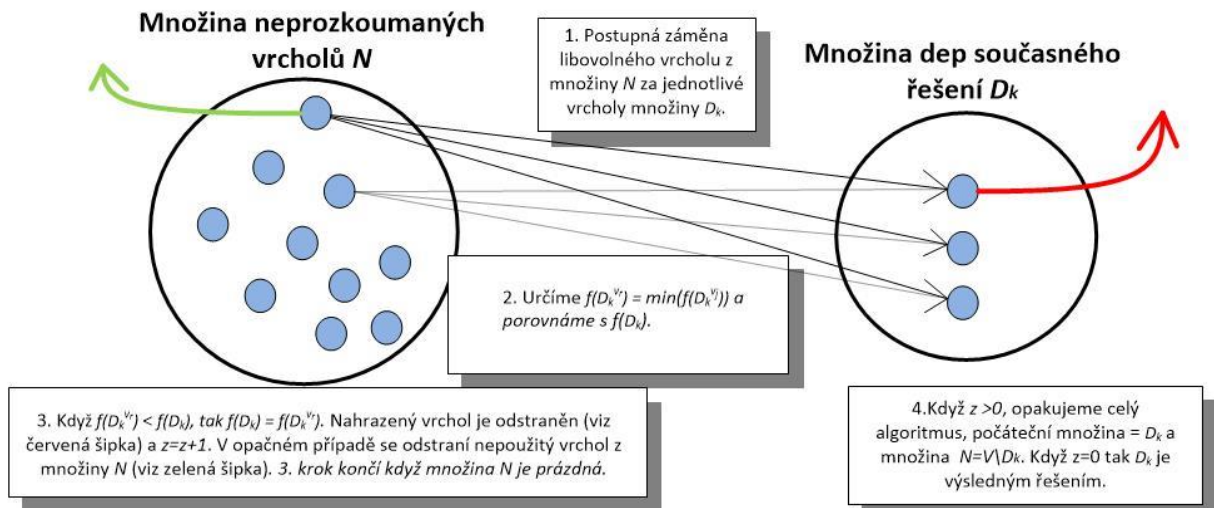
Mezi prosté heuristické metody patří i záměnná heuristika, kterou jsem zvolil pro řešení lokační úlohy. Záměnná heuristika se realizuje formou iterativního algoritmu.

Algoritmus se používá pro určení vrcholově a hranově optimální lokace  $k$  dep na síti a je založen na jednoduchém principu. Ten spočívá v zaměňování vrcholů aktuálního řešení vrcholy z množiny neprozkoumaných vrcholů  $N$ . V případě, že nové řešení dosahuje nižší hodnoty dopravní práce  $f(D_k)$ , přijme ho iterativní algoritmus jako nové aktuální řešení. Jestliže se tak nestane, zůstává řešení původní. Algoritmus probíhá do té doby, dokud se řešení mění (mění se pomocná proměnná  $z^{25}$ ) a dokud se neprozkoumají všechny vrcholy množiny  $N$ .

---

<sup>24</sup> Řešení TSP.

<sup>25</sup> V programování se jedná o tzv. „počítadlo“, které po splnění podmínky if přičte 1.



Obr. 3: Popis iterativního algoritmu [autor]

Algoritmus se skládá z pěti základních kroků a obecně lze zapsat následovně [16]:

### 1. krok:

- Zvolíme výchozí množinu dep  $D_k \subset V, k = |D_k|$ ,
- určíme množinu neprozkoumaných vrcholů  $N = V \setminus D_k$  (rozdíl množin  $V$  a  $D_k$ ),
- položíme pomocnou proměnou  $z = 0$  (registrace změn množiny  $D_k$ ),
- určíme  $f(D_k)$ .

### 2. krok:

- Zjistíme, zda je množina neprozkoumaných vrcholů prázdná:
  - **2a:** Je-li  $N = \emptyset$ , pokračujeme krokem 4,
  - **2b:** je-li  $N \neq \emptyset$ ,
    - Vybereme libovolný  $v \in N$  a vytvoříme množiny
 
$$D_k^{v_j} = D_k - \{v_j\} + \{v\}, \text{ pro } \forall v_j \in D_k,$$
    - vypočteme  $f(D_k^{v_j})$ ,
    - určíme  $f(D_k^{v_r}) = \min_{v_j \in D_k} \{f(D_k^{v_j})\}$ .

### 3. krok:

- Porovnáváme hodnoty kritéria:
  - **3a:** Pokud  $f(D_k^{v_r}) \geq f(D_k)$ , položíme  $N = N - \{v\}$  a pokračujeme krokem 2,
  - **3b:** pokud  $f(D_k^{v_r}) < f(D_k)$ , vytvoříme novou množinu dep  $D_k = D_k - \{v_r\} + \{v\}$ , položíme  $z = z + 1$  a  $f(D_k) = f(D_k^{v_r})$  a pokračujeme krokem 2.



#### 4. krok:

- **4a:** Je-li  $z = 0$ , pokračujeme krokem 5,
- **4b:** je-li  $z > 0$ , položíme znovu  $z = 0$ , určíme novou množinu  $N = V \setminus D_k$ , pokračujeme krokem 2.

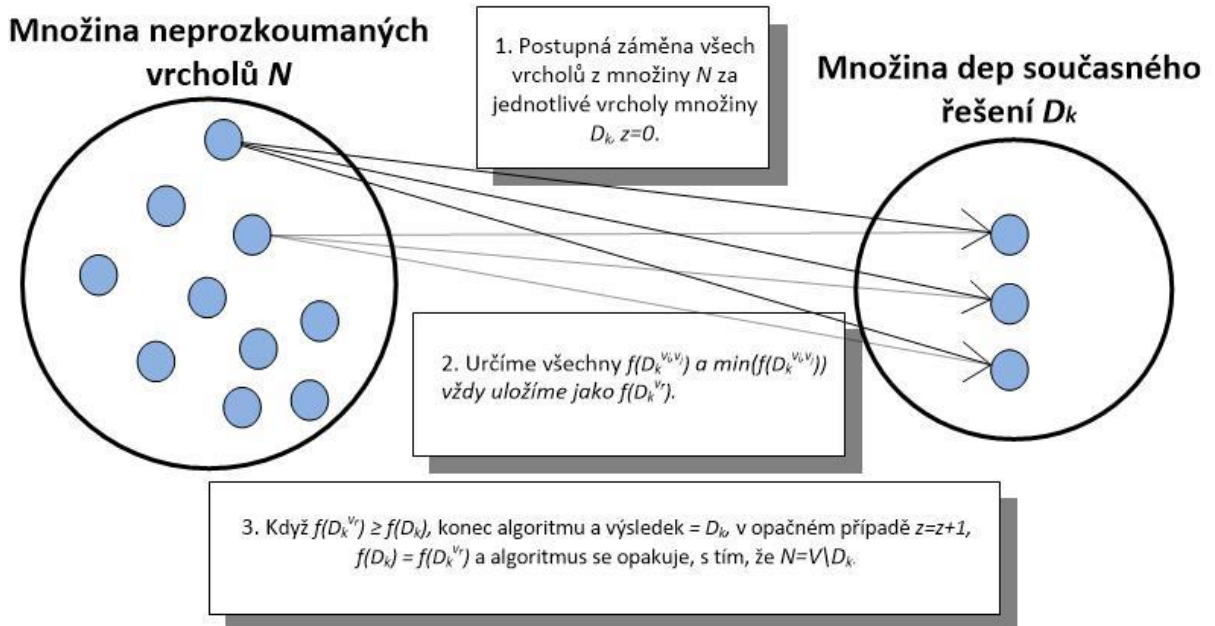
#### 5. krok:

- Množina  $D_k$  představuje vrcholově (sub)optimální rozmístění  $k$  dep na síti. Hodnota  $f(D_k)$  je minimální hodnota dopravní práce, která může být dosažena pomocí iterativního algoritmu při zvolené počáteční množině dep.

#### 3.2.2.6 Upravený iterativní algoritmus

Z klasického iterativního algoritmu, jde malou změnou vytvořit odlišně fungující iterativní algoritmus. U klasického iterativního algoritmu, se postupně nahrazují vrcholy z neprozkoumané množiny  $N$  do množiny  $D_k$  a hledáme minimální hodnotu účelové funkce (dopravní práce). Jeli nová minimální hodnota účelové funkce menší než účelová funkce současného řešení, ihned dosadíme daný vrchol z  $N$  do  $D_k$ . Nahrazený vrchol je odstraněn a nefiguruje v tuto chvíli v žádné z množin. Jestliže vrchol z  $N$  není vložen do současného řešení je poté sám odstraněn z  $N$  (viz obr. 3).

V upraveném iterativním algoritmu je princip velice podobný, avšak minimální dopravní práci a odpovídající vrcholy všech záměn z  $N$  do  $D_k$  si zapisujeme do proměnné stranou a procházíme všechny kombinace v jednom kroku. Dopravní práce jednotlivých kombinací se v tomto případě neznačí  $f(D_k^{v_j})$ , ale  $f(D_k^{v_i, v_j})$ , kde  $v_i$  představuje aktuálně nahrazovaný vrchol z množiny  $N$  a  $v_j$  nahrazený vrchol z množiny  $D_k$ . Po provedení všech záměn vrcholů z  $N$  s množinou aktuálního (současného) řešení  $D_k$  se minimální nalezená hodnota dopravní práce  $f(D_k^{v_r})$  porovnává s hodnotou dopravní práce současného řešení  $D_k$ . Dojde-li k nalezení lepšího řešení, přepíše se množina  $D_k$  na novou námi nalezenou množinu a algoritmus se opakuje ( $z = z + 1$ ) s tím, že  $N = V \setminus D_k$ . V opačném případě je algoritmus ukončen ( $z = 0$ ) a výsledkem je řešení  $D_k$  s dopravní prací  $f(D_k)$ .

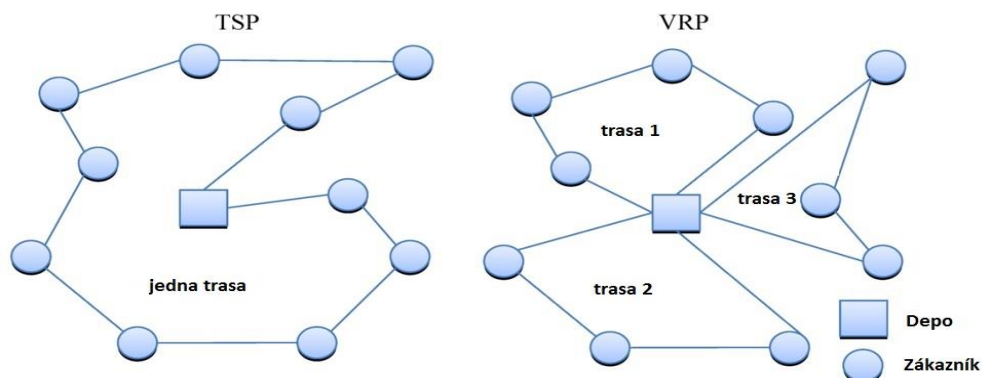


Obr. 4: Popis upraveného iterativního algoritmu [autor]

### 3.2.3 Okružní jízdy (Vehicle Routing Problem - VRP)

Metoda okružních jízd patří (stejně jako lokační analýza) k nejvíce studovaným kombinatorickým optimalizačním problémům a řadí se mezi tzv. NP-úplné (nedeterministicky polynomiální) problémy.

Jedná se o v praxi velice často využívané a obtížně řešitelné úlohy a zobecnění úlohy obchodního cestujícího (Travelling Salesman Problem - TSP). Úkolem TSP je nalézt nejkratší trasu, začínající a končící ve stejném místě, která zahrnuje všechna místa ze seznamu. Každé místo může být navštíveno pouze jednou. Oproti tomu VRP řeší stanovování optimálních cest (tras) používaného vozového parku, s cílem obsloužit požadavky zákazníků z jednoho nebo z více skladů. Nemusí se tedy používat pouze jeden dopravní prostředek, jako je tomu u TSP, ale ujetá vzdálenost se optimalizuje použitím více vozidel a vytvořením více tras (viz obr. 5).



Obr. 5: Porovnání TSP a VRP [12]

V úlohách okružních jízd se objevuje mnoho omezujících podmínek či rozšíření, které jednotlivé VRP úlohy navzájem odlišují. Úlohy se liší například tím, že rozvoz obsahuje jak dodávku k zákazníkům tak i dodávku od zákazníka do jiného skladu, každá trasa může být omezena maximální možnou délkou jízdy, použitím heterogenního vozového parku, časovým omezením a časovými okny, více násobnou obsluhou skladů, použitím více centrálních skladů, omezeným počtem automobilů nebo použitím jednoho automobilu na více než jedné trase, pokud čas nepřekročí stanovenou hodnotu atd.

Mezi základní druhy VRP patří [3]:

- Vehicle Routing Problem with Pickup and Delivery (VRPPD): Úloha řešící vyzvednutí zboží mimo centrální sklad a jeho následné doručení k zákazníkovi,
- Vehicle Routing Problem with Time Windows (VRPTW): Úloha s definovanými časovými okny, ve kterých je potřeba daný sklad navštívit,
- Capacited Vehicle Routing Problem (CVRP): Vozidla mají omezenou kapacitu, kterou nelze překročit,
- Vehicle Routing Problem with Multiple Trips (VRPMT): Vozidla mohou být použita na více než jedné trase (při respektování časové podmínky, existuje-li),
- Open Vehicle Routing Problem (OVRP): Vozidla se nemusí vracet na místo, odkud vyjela.

Obecná úloha okružních jízd je známá jako CVRP (Capacited Vehicle Routing Problem). CRVP počítá s deterministickými dodávkami (jsou předem známé a nelze je dělit), všechny vozidla mají stejnou kapacitu  $c$ , existuje kapacitní omezení automobilů a je využíván pouze jeden centrální sklad v místě  $v_0$  a  $n$  pobočných skladů  $v_1, v_2, \dots, v_n$ . V úloze jsou známy vzdálenosti  $d_{ij}; i, j = 1, 2, \dots, n$ , mezi libovolnou dvojicí obsluhovaných skladů  $v_i$  a  $v_j$ . Cílem je minimalizovat celkové náklady (počet tras nebo jejich délky a cestovní doby) [1]. V úloze platí, že cena dopravy mezi libovolnou dvojicí obsluhovaných skladů  $v_i$  a  $v_j$  je stejná v obou směrech, tzn., že výsledná matice nákladů je symetrická [16].

Z definice CVRP je patrné, že se jedná o zjednodušený model úlohy. V praxi se lze setkat častěji s jinými typy VRP. Základní model úlohy se dá efektivně řešit například pomocí algoritmu Clarka a Wrighta. Algoritmus není v diplomové práci popsán, neboť nebyl k výpočtu úlohy použit. Důvod je ten, že okružní jízdy řešené v praktické části diplomové práce spadají do úloh typu VRPTW, které jsou složitější než CVRP. Například bylo zapotřebí použít automobily různých kapacit a distribuci ze tří skladů. K výpočtům byl použit freeware, který pracuje se speciální sadou nástrojů (JSPRIT) v programovacím jazyku Java. Výpočty

VRPTW softwaru jsou založené na výpočtu LNS (Large Neighborhood Search) prováděném s pomocí meta-heuristické metody simulované žíhání. V následujícím odstavci je uveden a popsán model úlohy VRPTW jako model lineárního programování.

### 3.2.3.1 Model VRPTW jako model lineárního programování

Účelová funkce:

$$\min \sum_{r \in R} \sum_{i \in D} \sum_{\substack{j \in D \\ i \neq j}} d_{ij} x_{ijr} \quad (17)$$

Za podmínek:

$$\sum_{r \in R} \sum_{\substack{i \in D \\ i \neq j}} x_{ijr} = 1 \quad \text{pro } j \in J \quad (18)$$

$$\sum_{j \in J} x_{Sjr} \leq 1 \quad \text{pro } r \in R \quad (19)$$

$$\sum_{\substack{i \in D \\ i \neq j}} x_{ijr} = \sum_{\substack{i \in D \\ i \neq j}} x_{jir} \quad \text{pro } j \in D, r \in R \quad (20)$$

$$\sum_{j \in J} b_j \sum_{\substack{i \in D \\ i \neq j}} x_{ijr} \leq K_r \quad \text{pro } r \in R \quad (21)$$

$$t_i^r + t_{ij} \leq t_j^r + t_{max}(1 - x_{ijr}) \quad \text{pro } r \in R, i \in J, j \in J, i \neq j \quad (22)$$

$$t_i^r \leq h_j \quad \text{pro } r \in R, j \in J \quad (23)$$

$$t_j^r \geq d_j \quad \text{pro } r \in R, j \in J \quad (24)$$

$$x_{ijr} \in \{0,1\} \quad \text{pro } r \in R, i \in D, j \in D, i \neq j \quad (25)$$

kde:

$J$  = množina vrcholů dopravní sítě  $D = J \cup \{S\}$ ,

$S$  = vrchol, ve kterém se nachází sklad,

$b_j$  = zákazníkem  $j \in J$  požadovaný počet jednotek zboží,

$\langle d_j, h_j \rangle$  = definované časové okno,

$R$  = množina vozidel,

$d_{i,j}$  = známá vzdálenost libovolné dvojice vrcholů  $i, j$  z  $D$ ,

$t_{i,j}$  = kladná doba přesunu mezi uzly  $i, j$  z  $D$ .

Podmínka (18) určuje, že každý zákazník je obsloužen pouze jednou. Podmínka (19) zabezpečuje, že každé vozidlo bude použito jednou. Podmínka (20) vyjadřuje 1. Kirchhoffův zákon. Podmínka (21) zabezpečuje nepřekročení kapacity  $r$  – tého vozidla a podmínka (22), (23) a (24) zabezpečuje obsloužení zákazníků ve stanovených časových oknech. Podmínka (25) určuje, dojde-li k přiřazení  $r$  – tého vozidla na trasu  $i, j$  nebo nikoliv [10].

### 3.2.3.2 Simulované žíhání (Simulated Annealing – SA)

Před kapitolou okružních jízd, byly uvedeny pouze exaktní a heuristické metody. Existují ale i meta-heuristické metody. Velikou výhodou meta-heuristik oproti prostým heuristickým metodám je schopnost opustit, neboli vybědnout z lokálního minima a přejít pomocí iterativních kroků k řešení jinému. Klasická heuristika není schopná opustit lokální minimum poté, co ho předchozími kroky dosáhla (například výše zmíněná záměnná heuristika). Mezi základní charakteristiku meta-heuristiky se považuje seznam přípustných operací. Pomocí tohoto seznamu se definuje přípustných přechod mezi dvěma řešeními úlohy [7]. Mimo simulovaného žíhání do meta-heuristik patří například metoda tabu search a genetické algoritmy.

Meta-heuristická metoda simulované žíhání pracuje na velmi podobném principu jako minimalizační klasické heuristiky. Také totiž obsahuje strategii první vhodný a postupně prohledává okolí současného řešení. Pokud nalezne vhodný přechod, tedy takový, který vede ke snížení účelové funkce, tak ho realizuje [7].

Odlišnost od klasických heuristik nastává ve chvíli, kdy přechod není vhodný. Simulované žíhání tento přechod nezamítne, ale o realizaci přechodu se rozhoduje dle výsledku náhodného experimentu. Takovýto experiment se provádí s určitou pravděpodobností. Pravděpodobnost  $p(x, x^i, t) = e^{-(f(x) - f(x^i))/t}$  je ve prospěch přechodu a doplňková pravděpodobnost  $1 - p(x, x^i, t)$  proti němu. Z tohoto důvodu je zřejmé, že simulované žíhání umožňuje přechod od současného řešení  $x^i$  k následnému řešení  $x^{i+1}$ , které může mít i horší hodnotu účelové funkce. Zároveň je důležité vědět, že pravděpodobnost, se kterou se nevhodný přechod uskuteční, je tím menší, čím větší je zhoršení účelové funkce způsobené přechodem. Velice důležitý parametr, který také hraje roli v tom, jestli se přechod uskuteční či nikoliv je teplota. Teplota se značí písmenem  $t$ . Čím je teplota vyšší, tím je vyšší i pravděpodobnost uskutečnění přechodu. Parametr  $t > 0$  je na počátku meta-heuristiky nastaven na svojí maximální hodnotu  $t^{max}$  a postupně se během výpočtu snižuje (například po zvoleném počtu  $q$  prozkoumaných přechodů). Na začátku metody je tedy pravděpodobnost přechodu i k řešení s horší účelovou funkcí nejvyšší a podporuje se tak možnost opustit lokální minimum [7].

Snižování teploty po určitém počtu prozkoumaných řešení se určuje dle následujícího vztahu:

$$t = \frac{t}{1 + \beta * t}; \beta > 0 \quad (26)$$

Postupným snižováním parametru  $t$  se docílí ustálení se výpočtu v obecném lokálním minimu, kde výpočet končí definovaným pravidlem zastavení. Mezi běžně používané pravidlo patří ukončení výpočtu, které nastane nenalezením nového řešení během  $n$  kroků. Po celou dobu výpočtu se zapisuje nejlepší nalezené řešení  $x^*$ , které je po skončení algoritmu výstupem.

Jednotlivé kroky simulovaného žihání pro úlohu TSP jsou následovné [7]:

0. Krok: Pro počáteční řešení dané cyklickým seznamem  $x^0$  se definuje dosud nejlepší nalezené řešení  $x^* = x^0$ , položí se  $i = 0$  a teplota se nastaví na  $t = t^{max}$ .
1. Krok: Inicializace počtu aktualizací dosud nejlepšího nalezeného řešení od posledního zahřátí  $v = 0$ , inicializace počtu prozkoumaných přechodů od přechodu k současnému řešení  $r = 0$  a inicializace celkového počtu prozkoumaných přechodů od poslední změny teploty  $w = 0$ .
2. Krok: Vybrání řešení  $x$  z okolí  $O(x^i)$  a položení  $w = w + 1$ ,  $r = r + 1$   
Je-li  $w = q$ , tak  $t = \frac{t}{1+\beta t}$  a  $w = 0$ .
3. Krok: Je-li  $f(x) \leq f(x^i)$ , a  $i = i + 1$ ,  $x^i = x$ ,  $r = 0$  a  
Je-li  $f(x^i) < f(x^*)$ , tak  $x^* = x^i$ ,  $v = v + 1$  a přejde se na 5. krok.
4. Krok (Pokus o nevhodný přechod):  
Výpočet  $p(x, x^i, t) = e^{-(f(x)-f(x^i))/t}$  a vygenerování hodnoty  $h$  náhodné proměnné s rovnoměrným rozdělením pravděpodobnosti z intervalu  $(0, 1)$ . Je-li  $h \leq p(x, x^i, t)$ , tak  $i = i + 1$ ,  $x^i = x$ ,  $r = 0$  a přejde se na 5. krok.
5. Krok: Je-li  $r = u$ , tak se přejde na 6. Krok, jinak na 2. Krok.
6. Krok: Je-li  $v > 0$ , položte  $t = t^{max}$  {zahřívání} a přejděte na 1. Krok, jinak SA končí.

### 3.3 Vícekriteriální rozhodování

V kapitole, zabývající se matematickým programováním, byly uvedeny příklady s jedním optimalizačním kritériem, neboli s jednou účelovou funkcí. Při reálném rozhodování je třeba počítat i s více optimalizačními kritérii. Tato kritéria bývají zpravidla ve vzájemném nesouladu. To znamená, že například varianta nejlépe hodnocená dle jednoho kritéria nebude nejlépe hodnocena dle jiného kritéria. Vícekriteriální přístup aplikujeme v běžném životě jako rozhodovací proces s jedním racionálním rozhodovatelem. Pro rozhodnutí bereme v úvahu více často protichůdných hledisek – hledáme kompromisní řešení. Cílem při analýze vícekriteriálního rozhodování je řešit konflikt mezi vzájemně protikladnými kritérii a konkrétním cílem je výběr z jedné varianty, která bude podkladem pro konečné rozhodnutí [6].

Obecně bývá vícekriteriální rozhodování oproti monokriteriálnímu složitější, pracnější a podléhá většímu vlivu subjektivity. Oproti tomu v monokriteriálním rozhodování je nebezpečí přílišného zjednodušení, tedy zanedbání méně důležitých hledisek. Tento fakt nazýváme tzv. násilnou agregací. Mezi hlavní subjektivní faktory vícekriteriálního rozhodování patří [23]:

- Výběr hodnotících hledisek,
- kvantifikace důležitosti hodnotících hledisek,
- výběr metody, parametru výpočtu.

Subjektivní faktory se dají eliminovat použitím speciálních technik. Mezi ně patří například citlivostní analýza, expertní metody nebo obecně dodržování standardních postupů.

Rozdělení vícekriteriálního rozhodování [6]:

- **Vícekriteriální hodnocení variant (VHV):** analýza rozhodovacích problémů s konečným počtem variant zadaným ve formě seznamu, které jsou hodnoceny podle několika kritérií,
- **vícekriteriální programování, vícekriteriální lineární programování (VLP):** analýza vícekriteriálních rozhodovacích problémů, ve kterých je množina přípustných řešení definována pomocí soustavy omezujících podmínek jako v úlohách matematického programování.

Vícekriteriální hodnocení variant je rozhodovací proces s konečnou množinou variant přípustných řešení  $V = \{V_1, V_2, \dots, V_n\}$  a více kritérii optimality  $K_1, K_2, \dots, K_k$ . Každá z variant je popsána vektorem kritériálních hodnot. Úloha VHV se může vyjadřovat ve tvaru tzv. kritériální matice.

Kvalita výsledků úloh vícekriteriálního rozhodování je silně závislá na souboru kritérií. Proto je nutné definovat požadavky na tento soubor. Mezi hlavní požadavky patří úplnost souboru kritérií, rozumný počet kritérií, neredundantnost<sup>26</sup> kritérií. V poslední řadě musí být kritéria relevantní, jednoznačně určená a číselně reálná.

Možnosti využití vícekriteriálního hodnocení variant jsou velké. Hlavním důvodem jejich častého využívání v praxi, může být i to, že VHV je srozumitelné v podstatě pro každého. Mezi příklady použití VHV v praxi patří [6]:

- Výběr lokality pro realizaci investiční akce,

---

<sup>26</sup> Neopakování.

- konkurzní přijímací řízení,
- hodnocení vospělosti země,
- hodnocení výrobků či služeb.

Ne vždy je cílem VHV výběr pouze jedné varianty. Často jde rozhodovateli například o uspořádání variant (od nejlepší po nejhorší) nebo o klasifikaci variant. Klasifikací variant se myslí jejich rozdělení do určitých kategorií. Například u řešení konkurzního přijímacího řízení může jít o dvě skupiny - přijetí a nepřijetí [6].

### 3.3.1 Maximalizační a minimalizační typ kritérií

Kritéria VHV mohou být buď minimalizačního nebo maximalizačního typu. U maximalizačních kritérií jsou lépe hodnoceny varianty s vyššími hodnotami a u minimalizačního typu je tomu naopak. Protože některé metody VHV vyžadují kritéria stejného typu (většinou maximalizačního typu) je potřeba znát vzorce pro jejich převod. Pevod se liší u jednotlivých stupnic měření a proto je nutné jednotlivé škály popsat.

První stupnicí měření je **nominální škála**. Stupnice popisuje různost či rovnost hodnot a v praxi se může jednat například o barvu, pohlaví atd. Další stupnicí a zároveň tou nejpoužívanější je **pořadová (ordinální) škála**. Ta řeší otázku „menší či větší“ a může se jednat například o Mohsovu stupnici tvrdosti nerostů. Naopak **intervalová škála** popisuje o kolik menší či větší jsou hodnoty variant. Posledním typem kritérií dle stupnice měření jsou kritéria zapsána v **poměrové škále** (kolikrát menší či větší jsou hodnoty variant).

#### Pevod minimalizačních kritérií na maximalizační

1) Kvalitativní kritéria (stačí zachovat pouze informaci „stejný, lepší, horší“ a proto se převod často řeší jednoduchým otočením bodovací stupnice) [23]:

$$f'(x_i) = f_{max} + f_{min} - f(x_i), \quad (27)$$

kde  $f'(x_i)$  je hodnota upraveného maximalizačního kritéria,  $f(x_i)$  je hodnota původního min. kritéria,  $f_{max}$  je maximální hodnota bodové stupnice a  $f_{min}$  je minimální hodnota bodové stupnice.

2) Kvantitativní kritéria měřená v intervalové stupnici

$$f'(x_i) = \max_j f(x_j) + \min_j f(x_j) - f(x_i) \quad (28)$$



3) Kvantitativní kritéria měřená v poměrové stupnici

$$f'(x_i) = \frac{\max_j f(x_j) \min_j f(x_j)}{f(x_i)} \quad (29)$$

### 3.3.2 Normování hodnot kritérií

Normování jednotlivých hodnot kritérií se provádí kvůli jejich převodu na sčitatelné (bezrozměrné) veličiny. Normované hodnoty nabývají hodnot z intervalu  $< 0; 1 >$ . Normování se provádí odlišně pro kritéria měřená v různých stupnicích stejně jako v případě převodu z minimalizačních kritérií na maximalizační [23].

1) Kvalitativní kritéria (požadujeme pouze, aby nejmenší hodnotě bodové stupnice odpovídala normovaná hodnota 0 a nejvyšší hodnotě bodové stupnice hodnota 1)

$$f'(x_i) = \frac{f(x_i) - f_{min}}{f_{max} - f_{min}}, \quad (30)$$

kde  $f'(x_i)$  je normovaná hodnota kritérií,  $f(x_i)$  je původní hodnota kritéria,  $f_{max}$  je maximální hodnota bodové stupnice a  $f_{min}$  je minimální hodnota bodové stupnice.

2) Kvantitativní kritéria měřená v intervalové stupnici

$$f'(x_i) = \frac{f(x_i) - \min_j f(x_j)}{\max_j f(x_j) - \min_j f(x_j)} \quad (31)$$

3) Kvantitativní kritéria měřená v poměrové stupnici

$$f'(x_i) = \frac{f(x_i)}{\max_j f(x_j)} \quad (32)$$

### 3.3.4 Metody stanovení vah kritérií

Získání vah kritérií může být obtížný problém. Aby se hodnotiteli usnadnila práce, stanovují se váhy podle jednoduchých postupů. Metody stanovení vah kritérií jsou nástroje, které pracují se subjektivními informacemi od rozhodovatele a určují odhady vah kritérií. Váhy kritérií se značí  $v_k$  ( $k = 1, \dots, m$ ) a vyjadřují relativní důležitost kritérií. Obvykle se pracuje s normovanými vahami kritérií, pro které platí následující vztahy [23]:

$$\sum_{k=1}^m v_k = 1; v_k \geq 0 \quad (33)$$

#### 1. Přímá metoda

Hodnotitel stanovuje normované váhy kritérií v souladu se svými představami. Metoda se vyznačuje svojí jednoduchostí. Na druhou stranu je nutno kontrolovat součet a při větším množství kritérií se metoda stává nepřehlednou.

## 2. Bodovací metoda

Důležitost kritérií vyjadřuje rozhodovatel číslem ve vhodně definované bodovací stupnici. Obecně se doporučuje, aby stupnice začínala od 0 (zcela nedůležité kritérium) a měla lichý počet stupňů. Použitím neceločíselného hodnocení můžeme docílit zjemnění hodnotící stupnice. Vztah pro normování takto získaných vah je následovný [23]:

$$v_k = \frac{b_k}{\sum_{i=1}^m b_i}, \quad (34)$$

kde  $b_k$  vyjadřuje počet bodů přiřazených  $k$  – tému kritériu.

## 4. Fullerova metoda (metoda párového srovnání)

Metoda je založena na porovnávání důležitostí každé dvojice kritérií a postupuje se následovně [23]:

- $b_{i,j} = 1$  pokud je  $i$  – té kritérium důležitější než  $j$  – té,
- $b_{i,j} = 0$  pokud je  $i$  – té kritérium méně důležité než  $j$  – té,
- $b_{i,j} = 0,5$  pokud je  $i$  – té a  $j$  – té kritérium stejně důležité.

Mělo by platit, že  $b_{i,j} + b_{j,i} = 1$  pro  $i \neq j$ ;  $b_{i,i} = 0$ .

Normované váhy se počítají dle vztahu:

$$v_k = \frac{2 \sum_{j=1}^m b_{k,j}}{m(m-1)} \quad (35)$$

## 5. Metoda pořadí

Metoda pořadí je speciálním případem metody bodovací. Hodnotitel jednoduše seřadí kritéria podle důležitosti (připouští se rovnocennost) [23].

- $b_k$  je hodnota pořadové funkce.
  - Nejdůležitější kritérium  $b_k = m$ ,
  - nejméně důležité kritérium  $b_k = 1$ ,
  - pro rovnocenná kritéria se počítá sdružené (průměrné) pořadí.

Normování je stejné jako u bodovací metody, tedy ve tvaru:

$$v_k = \frac{b_k}{\sum_{i=1}^m b_i} = \frac{2b_k}{m(m+1)} \quad (36)$$

## 6. Saatyho metoda

Stejně jako v případě Fullerovy metody, i zde dochází k porovnávání všech možných dvojic kritérií. Saatyho metoda je ovšem o něco propracovanější a je ze všech metod stanovení vah kritérií nejpoužívanější. Stanovená míra preference mezi všemi dvojicemi kritérií se zapisuje do tzv. Saatyho matice  $S = (s_{ij}; i, j = 1, 2, \dots, k)$  a hodnotitel zde stupeň důležitosti jednoho kritéria před druhým vyjadřuje pomocí stupnice od 1 do 9, kde hodnota 1 odpovídá tomu, že dvojice kritérií má stejnou důležitost. Stejně jako ve Fullerově metodě, i v Saatyho metodě

stačí spočítat pouze neuspořádané dvojice a prvky symetrické podle diagonály dopočítat dle vztahu [6]:

$$s_{j,i} = \frac{1}{s_{i,j}} \quad (37)$$

Normované váhy poté stanovíme pomocí iteračního algoritmu s rychlou konvergencí.

$$h^{(0)} = (1, 1, \dots, 1) \quad (38)$$

$$v_j^{(k)} = \frac{h_j^{(k)}}{\sum_{i=1}^m h_i^{(k)}} \quad (39)$$

$$h^{(k+1)} = S * v^{(k)} \quad (40)$$

### 3.3.5 Metody vícekritériálního hodnocení variant

V současné době existuje velké množství metod pro VHV, které jsou založené na různých principech. Základní dělení metod je na jednoduché metody, metody založené na párovém srovnávání variant, interaktivní metody a metody založené na prazích citlivosti. Jmenovitě se jedná například o metody třídy ELECTRE, PROMETHEE, metodu globálního kritéria či metody váženého součtu pořadí. Podrobněji budou popsány pouze některé z tzv. jednoduchých metod [6].

Mezi jednoduché metody patří [23]:

- Lexikografická metoda,
- metoda váženého součtu pořadí,
- metoda globálního kritéria,
- metoda bazické varianty,
- metoda PATTERN,
- cílové programování,
- TOPSIS,
- vícekritériální funkce utility.

#### 1. Lexikografická metoda (postupné diktatury)

Lexikografická metoda je založena na seřazení kritérií podle důležitosti a následném vybrání optimální varianty dle nejdůležitějšího kritéria. Nastane-li případ, kdy je optimálních variant podle kritéria prvního v pořadí více, vybíráme optimální variantu dle kritéria druhého v pořadí atd. Nevýhodou LM je to, že i při značném uvolnění diktatury se řada kritérií vůbec neuplatní [23].

## 2. Metoda váženého součtu pořadí

V metodě váženého součtu pořadí se hodnoty kritérií  $f_{i,j}$  převádí na pořadovou funkci  $q_{i,j}$  následovně [23]:

- $q_{i,j} = 1$  pro nejlepší variantu,
- $q_{i,j} = n$  pro nejhorší variantu,
- u rovnocenných variant používáme sdružené (průměrné) pořadí.

Optimální varianta je poté vybírána na základě kritéria:

$$\min_i \sum_{j=1}^m v_j q_{i,j} \quad (41)$$

## 3. Metoda globálního kritéria (metoda váženého součtu)

MGK předpokládá všechny kritéria maximalizačního typu. Je tedy nutné všechna minimalizační kritéria převést pomocí k tomu určených vztahů. Dále je zapotřebí počítat s normovanými hodnotami kritérií. Principem metody je maximalizace váženého součtu hodnot kritérií:

$$\max_i \sum_{j=1}^m v_j f_{i,j} \quad (42)$$

## 4. Metoda bazické varianty

V názvu této metody se nachází pojem bazická varianta a myslí se jím porovnávací varianta, která se používá během výpočtu. Nejčastěji se za porovnávací variantu volí varianta průměrná či s nejlepšími hodnotami. Stejně jako u metody předchozí, je zapotřebí pracovat pouze s maximalizačním typem kritérií. Je-li tato podmínka splněna, můžeme optimální variantu vybrat dle následujícího vztahu:

$$\max_i \sum_{j=1}^m v_j z_{i,j}; \quad z_{i,j} = \frac{f_{i,j}}{f_{b,j}}, \quad (43)$$

kde  $f_{b,j}$  je hodnota  $j$  – tého kritéria pro bazickou variantu a  $z_{i,j}$  je tzv. míra dosažení vlastní bazické varianty.

### 3.4 Teorie zásob

Poslední disciplína operačního výzkumu, která byla použita k výpočtu řešeného problému, je teorie zásob. Na začátku kapitoly je důležité definovat, co zásoby vlastně jsou. Zásobami nazýváme předměty, které se uchovávají na pozdější spotřebu neboli nevyužitý zdroj sloužící ke krytí budoucí spotřeby. Jiná definice může zásoby uvádět jako okamžitě použitelné zdroje, které jsou systematicky vytvářeny k materiálovému zabezpečení plynulého průběhu výrobního procesu či uspokojení poptávky na trhu [11].

Teorie zásob se zabývá řízením zásobovacího procesu a optimalizací objemu skladovaných zásob s ohledem na minimalizaci nákladů, případně ztrát, které souvisejí s udržováním, objednáváním a vydáváním zásob ze skladu [6]. Jde o relativně mladou disciplínu. Dříve se problémy zásob řešily pomocí subjektivních a empirických metod a až po druhé světové válce se začaly mimo deterministické teorie zásob používat i stochastické metody.

Společnosti mívají často v zásobách vázanou velkou část svých aktiv. Optimalizace a řízení zásob přispívá k částečnému uvolnění takto vázaných prostředků a navíc vede ke snížení nákladů spojených s probíhajícími zásobovacími procesy.

Mezi dvě hlavní otázky v souvislosti s řízením zásob patří [11]:

1. Kdy objednávat novou dodávku zboží?
2. Jak velká má být objednávka?

### **3.4.1 Základní pojmy teorie zásob**

**Substrátem** se v teorii zásob myslí vše, co se ve skladech může skladovat. Substrát, který do skladu vstupuje, nazýváme **dodávkami** a substrát vystupující ze skladu se nazývá **výdaje** či **spotřeba**.

Model teorie zásob se dle [11] skládá ze tří hlavních částí:

- Vstupu,
- skladování,
- výstupu.

**Mechanismus vstupu** představuje soubor pravidel, podle kterého přicházejí dodávky na sklad. Dodávky lze rozdělit na čtyři základní typy [11]:

1. Deterministické dodávky co do velikosti i času,
2. deterministické dodávky určené co do velikosti, ale náhodně rozložené v čase,
3. dodávky náhodně určené co do velikosti, ale deterministicky rozložené v čase,
4. dodávky náhodně určené co do velikosti i času.

**Mechanismus výstupu** představuje soubor pravidel, podle kterých se substráty spotřebovávají nebo jdou ve formě dodávek mimo podnik. Podle spotřeby rozdělujeme substrát na dva modely spotřeby a to modely se spojitou<sup>27</sup> a s diskretní spotřebou. S mechanismem výstupu souvisí také pojem deficit. Deficit je jinak řečeno vyčerpání zásob. V teorii zásob rozlišujeme systémy s deficitem a systémy s vyloučením deficitů. U systémů

---

<sup>27</sup> Častá, skoro kontinuální spotřeba. Neustálé spotřebovávání.

s odloženou spotřebou se neuspokojená spotřeba řeší z další dodávky. Naopak u systémů se ztracenou spotřebou nelze spotřebu řešit z další dodávky (potenciální prodej se ztrácí) [11].

Podstatnou částí modelů je kriteriální neboli účelová funkce. V teorii zásob se nejčastěji používá kriteriální funkce tzv. **funkce nákladů**. Ta se skládá ze tří částí a to nákladů na dodávku, nákladů na skladování a nákladů deficitu. Náklady na dodávku jsou náklady spojené s každou dodávkou. Naopak do nákladů na skladování patří mzdové a materiálové náklady spojené s udržováním zásob. Tyto náklady jsou závislé na množství zásob. Poslední část funkce nákladů jsou tzv. náklady deficitu, které mají rozdílný charakter u modelu s odkladem a u modelu se ztracenými prodeji. Modely budou podrobně popsány v kapitole 3.4.2. U výrobních podniků může mít vznik deficitu za následek přerušení výroby. Finanční ztráty způsobené přerušením<sup>28</sup> výroby lze také započítat do nákladové funkce. Další náklady mohou být v praxi různé sankce za nedodání materiálu apod. [11].

### **Motivace tvorby zásob**

Mezi tři hlavní motivace tvorby zásob patří transakční, predikční a spekulativní motivace. Ke každé z nich patří určitý druh zásob [13].

- Transakční motivace: jde o krytí spotřeby, která se vyskytuje trvale za normálních podmínek nebo průměrných podmínek na trhu. Tomuto účelu slouží běžná zásoba.
- Predikční motivace: jde o krytí předvídatelných sezónních výkyvů poptávky na trhu. Tomuto účelu slouží sezonní zásoba.
- Spekulativní motivace: jde o krytí náhodných výkyvů poptávky na trhu, které lze určit jen s jistou pravděpodobností. Tomuto účelu patří pojistná zásoba.

**Pojistná zásoba** je důležitým pojmem teorie zásob. Tato zásoba se vytváří proto, aby do požadované míry zachycovala náhodné výkyvy na straně vstupu a výstupu. V odborné literatuře lze naléznout pojem **signální zásoba** nebo **hladina objednání**, přičemž jde o objednávkový bod, který představuje takovou výši zásob, při které musí být provedena objednávka.

### **3.4.2 Deterministický a stochastický model zásob**

Modely zásob lze dělit z mnoha hledisek. Jmenovitě jde o rozdělení dle stacionárnosti spotřeby, spojitosti spotřeby, způsobu řešení čerpání zásob, zohlednění vývoje spotřeby v čase, počtu skladovaných substrátů nebo podle počtu skladů. Mimo těchto jmenovaných

---

<sup>28</sup> Může se jednat o vysoké částky (např. automobilový průmysl).

dělení existuje navíc zohlednění náhodných vlivů v modelu. Podle tohoto hlediska lze modely rozdělit na modely deterministické a stochastické [11].

## 1. Deterministické modely zásob

Deterministické modely zásob jsou založeny na úplné informovanosti a úplné ovladatelnosti. Úplnou informovaností rozumíme přesnou znalost všech základních prvků, jako jsou momenty dodávky, spotřeba, dodací lhůta, atd. Úplná ovladatelnost znamená, že systém, který řídí sklad, může dle potřeby určit velikost a časové rozdělení jednotlivých veličin. Deterministické modely zásob se používají především k řešení zidealizovaných úloh. Na úlohy v praxi, je použití těchto modelů do jisté míry omezené. Jednou z možností použití deterministických modelů na reálné příklady je počítání se středními hodnotami stochastických veličin.

Mezi základní druhy deterministickým modelů patří [11]:

- Model se spojitou spotřebou: spotřeba je spojitou lineární funkcí času,
- model s diskretní spotřebou: spotřeba se realizuje v ekvidistantních intervalech délky  $C_k$  po jednom kuse,
- model s konečnou intenzitou dodávky: dodávka se neuskutečňuje najednou, ale postupně. Intenzita dodávky  $d = \frac{q_{opt}}{C_d}$ ,  
kde  $q_{opt}$  je optimální velikost dodávky a  $C_d$  je období, během kterého se zásoby dodávají na sklad, případně vyrábějí.
- model s deficitem a odloženou spotřebou: deterministický model, který počítá i s možností, že nastane deficit.

## 2. Stochastické modely zásob

Jak bylo již řečeno, reálné úlohy se většinou nechovají tak, abychom je mohli řešit za pomoci deterministických modelů, a proto je třeba popsat modely stochastické. Jestliže řešená úloha obsahuje alespoň jeden stochastický faktor, kterým se myslí spotřeba, dodací lhůta, čas dodání nebo velikost dodávky, je zapotřebí použít stochastický model. Charakteristické pro stochasticky chovající se systémy je nelineární průběh spotřeby v grafu vývoje stavu zásob v závislosti na čase.

Stochastické modely se dělí na dvě základní skupiny. První skupinou jsou tzv. **dynamické stochastické modely**. Podle pravidla, kterým se určuje doplňování zásob (velikost dodávky, čas dodávky a čas zadání dodávky), dělíme dynamické stochastické modely na modely se signalizací změn a modely s periodickou kontrolou.

Prvně jmenované modely se signalizací změn pracují na principu, kdy je zaznamenávána každá změna zásob. Nová objednávka se zadává v případě, že stav zásob dosáhne tzv. hladiny objednání. U dynamických modelů s periodickou kontrolou se porovnává stav zásob s hladinou objednání pouze v určitých okamžicích. Tyto modely nachází uplatnění v systémech se složitou evidencí změn.

Mimo dynamické modely zásob existují i **stochastické statické modely**. Statické modely jsou charakteristické řešením zásob pouze během jednoho cyklu, a proto se zde konstruuje nákladová funkce jen pro jedno období. Protože stochastické statické modely zásob a dynamické stochastické modely s periodickou kontrolou nebyly v diplomové práci použity ve výpočtu řešeného problému, jsou dále popsány pouze dynamické modely se signalizací změn.

### 1. Model se signalizací změn s odloženou spotřebou

Model se signalizací změn s odloženou spotřebou se použije tehdy, když je spotřeba  $S(t)$  náhodná, spojitá a se střední hodnotou  $b$  jednotek za časovou jednotku, jeli dodací lhůta  $L$  náhodná, model je stacionární (tedy model, u kterého se rozdělení náhodných veličin v jednotlivých cyklech nemění), pokud se připouští deficit s odloženou spotřebou, velikost dodávky  $m$  se nemění v jednotlivých cyklech a dodávky se realizují celé najednou [11].

Cílem je získat dvě veličiny. Optimální velikost dodávky  $m_0$  a optimální hladinu objednání  $h_0$ . Obě veličiny se v modelu získávají pomocí nákladové funkce, kterou se snažíme minimalizovat. Protože je spotřeba náhodná, nelze  $h_0$  počítat přímo z velikosti dodávky.

Postup výpočtu modelu se signalizací změn s odloženou spotřebou je následovný [11]:

1. Výpočet střední spotřeby  $ES(L)$  během dodací lhůty  $L$

$$ES(L) = \int_0^{\infty} s f_s(s, L) ds, \quad (44)$$

kde  $f_s(s, L) ds$  se liší rozdělení spotřeby během dodací lhůty. Pokud se jedná o normální rozdělení, lze  $f_s(s, L)$  vypočítat následovně:

$$f_s(s, L) = \frac{1}{\sqrt{2\pi} \cdot \sigma} e^{-\frac{(s-\mu)^2}{2 \cdot \sigma^2}}, \quad (45)$$

kde  $\sigma$  je směrodatná odchylka a  $\mu$  střední hodnota normálního rozdělení. Rovnoměrné rozdělení je popsáno u modelu se signalizací změn a ztracenými prodeji.

2. Vzorec pro celkové náklady na skladování za časovou jednotku

$$q_2 \bar{m} = q_2 \frac{m+2h-2ES(L)}{2} \quad (46)$$



3. Určení střední hodnotu deficitu EU

$$EU = \int_h^\infty (s - h) f_s(s, L) ds = \int_h^\infty s f_s(s, L) ds - h F_s^*(h, L), \quad (47)$$

kde  $F_s^*(h, L)$  je doplňková distribuční funkce spotřeby S během dodací lhůty L (pro normální rozdělení se rovná:  $-h \int_{\frac{h-\mu}{\sigma}}^\infty \varphi(y) dy$  (musí zde být zavedena normovací substituce  $y = \frac{s-\mu}{\sigma}$ ).

4. Celkové náklady deficitu

$$q_4 n EU = q_4 \frac{b}{m} EU, \quad (48)$$

kde  $q_4$  jsou náklady závislé na velikosti deficitu. Jestliže sloučíme vzorce (47) a (48) se vzorcem pro náklady na dodávku za časovou jednotku  $q_1 n = q_1 \frac{b}{m}$ , kde  $q_1$  jsou náklady na jednu dodávku a  $b$  je střední spotřeba za časovou jednotku, dostává funkci nákladů ve tvaru:

$$H(m, h) = q_1 \frac{b}{m} + q_2 \frac{m+2h-2ES(L)}{2} + q_4 \frac{b}{m} EU, \quad (49)$$

kde  $q_2$  jsou náklady na skladování jednotkového množství substrátu za jednotku času.

5. Vzorce<sup>29</sup> pro hledané optimální hodnoty  $m_0$  a  $h_0$

$$m = \sqrt{\frac{2b(q_1 + q_4 EU)}{q_2}} \quad (50)$$

$$F_s^*(h, L) = \frac{q_2 m}{q_4 b} \quad (51)$$

Tyto dvě rovnice jsou řešeny iteračním algoritmem:

1. Položíme  $EU=0$  a vypočítáme  $m_1$ .  
Pomocí  $m_1$  spočítáme  $h_1$ .
2. Pomocí  $h_1$  vypočítáme  $m_2$ .  
Pomocí  $m_2$  vypočítáme  $h_2$ .
3. Pomocí  $h_{i-1}$  vypočítáme  $m_i$ .  
Pomocí  $m_i$  vypočítáme  $h_i$ .

Algoritmus se používá do té doby, dokud hodnoty  $m_{i-1}$  a  $m_i$ ,  $h_{i-1}$  a  $h_i$  se od sebe neliší o námi stanovenou hodnotu. Poté aktuální řešení  $m_i$  a  $h_i$  prohlásíme za optimální řešení modelu.

<sup>29</sup> Vzorce jsou získané z první derivace funkce nákladů položené nule.

6. Výpočet střední průměrné množství zásob  $\bar{m}$

$$\bar{m} = \frac{m+2h-2ES(L)}{2}, \quad (52)$$

kde  $m$  je velikost dodávky.

## 2. Model se signalizací změn a ztracenými prodeji

Všechny předpoklady uvedené u předchozího modelu platí i zde. Jediný rozdíl je řešení deficitu. V modelu se signalizací změn s odloženou spotřebou se deficit řeší následující dodávkou. U modelu se ztracenými prodeji se deficit neřeší vůbec, nebo se řeší z jiných zdrojů, což je doprovázeno zvýšením nákladů.

Naším cílem je opět získat dvě veličiny a to optimální velikost dodávky  $m_0$  a optimální hladinu objednání  $h_0$ .

Postup výpočtu modelu se signalizací změn a ztracenými prodeji je následovný [11]:

1. Výpočet střední spotřeby  $ES(L)$  během dodací lhůty  $L$

$$ES(L) = \int_0^{\infty} s f_s(s, L) ds, \quad (53)$$

kde  $f_s(s, L) ds$  pro rovnoměrné rozdělení pravděpodobnosti v intervalu  $\langle x_1; x_2 \rangle$  :

$$f_s(s, L) = \frac{1}{x_2 - x_1} \quad (54)$$

2. Určení střední hodnoty deficitu  $EU$

$$EU = \int_h^{\infty} s f_s(s, L) ds - h F_s^*(h) = \int_h^{x_2} s \frac{1}{x_2 - x_1} ds - h \left( 1 - \frac{h - x_1}{x_2 - x_1} \right) = \frac{h^2}{2(x_2 - x_1)} - k * h + c, \quad (55)$$

kde  $k$  a  $c$  jsou konstanty vzniklé integrací.

3. Sestavení funkce nákladů

$$H(m, h) = q_1 \frac{b}{m} + q_2 \frac{m+2h-ES(L)+2EU}{2} + q_4 EU \frac{b}{m} \quad (56)$$

4. Výpočet  $m_0$  a  $h_0$  za pomoci iteračního algoritmu (popsaného v předchozím modelu)

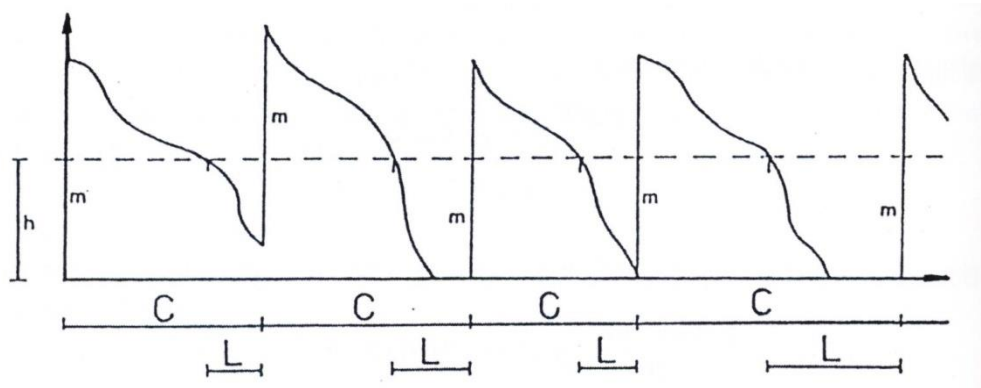
$$m = \sqrt{\frac{2b(q_1 + q_4 EU)}{q_2}} \quad (57)$$

$$F_s^*(h, L) = \frac{q_2 m}{q_4 b + q_2 m} \quad (58)$$

5. Výpočet střední průměrné množství zásob  $\bar{m}$  a průměrný počet dodávek  $n$ .

$$\bar{m} = \frac{m+h-ES(L)+EU+h-ES(L)+EU}{2} \quad (59)$$

$$n = \frac{b}{m_0} \quad (60)$$



Obr. 6: Stochastický model zásob se signalizací změn a ztracenými prodeji [11]

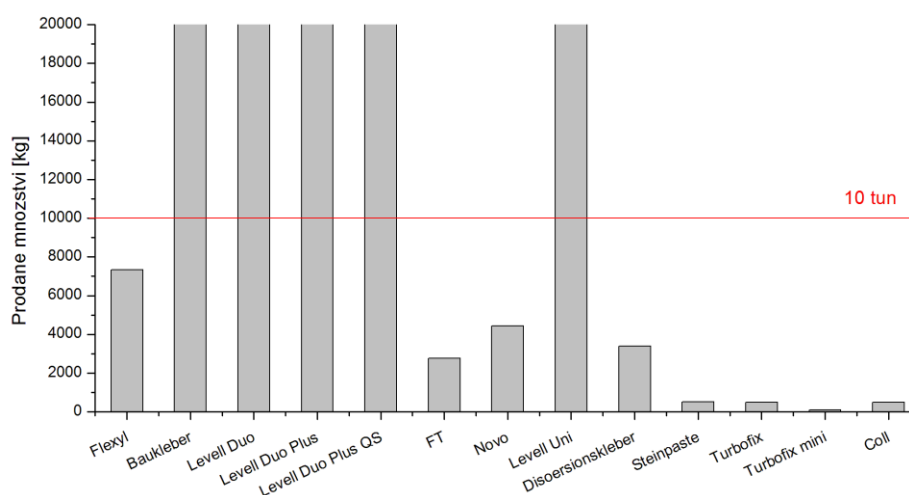
#### 4. Praktická část – Fáze výpočtu navrhovaných řešení

Čtvrtá kapitola se věnuje postupu výpočtu úlohy, který lze rozdělit na pět částí, respektive fází.

1. fáze: vícekriteriální rozhodování,
2. fáze: lokační analýza,
3. fáze: okružní jízdy s časovými okny (VRPTW),
4. fáze: stochastický model zásob,
5. fáze: ekonomické zhodnocení.

#### 0. fáze

Před samotným řešením úlohy bylo zapotřebí určit, s jakými druhy malt se bude počítat. Jako kritérium bylo zvoleno množství prodeje v roce 2014. Limit byl stanoven na 10 tun. Graf 3 znázorňuje druhy malt, které splnily kritérium. Jmenovitě jde o Levell Duo Plus, Levell Duo, Levell Uni, Baukleber a Levell Duo Plus QS.



Graf 3: Prodané množství malt společností Sto, s.r.o. v roce 2014 [autor]

Jednotlivé druhy malt se od sebe liší kvalitou, cenou a použitím. QS je označení pro maltu používanou na stavbách v zimních měsících. Pořadí obcí v tabulce 7 je sestaveno podle množství vybraných druhů malt, které do těchto míst firma Sto, s.r.o. dodala v roce 2014.

Pořadí obcí zůstává stejné během celého výpočtu a řídí se jím i číslování vrcholů dopravní sítě ve fázi lokační analýzy.

Tab. 7: Seřazení měst [autor]

1. Ostrava (114t)	10. Modřice (10t)	19. Tichá (3t)	28. Třanovice (3t)
2. Brno (109t)	11. Uničov (7t)	20. Uherské Hradiště (3t)	29. Heraltice (2t)
3. Havířov (94t)	12. Třinec (6t)	21. Zlín (3t)	30. Ivančice (2t)
4. Karviná (25t)	13. Hlučín (5t)	22. Břeclav (3t)	31. Kobylí (2t)
5. Olomouc (15t)	14. Třebíč (5t)	23. Klentnice (3t)	32. Znojmo (2t)
6. Luhačovice (12t)	15. Vsetín (5t)	24. Židlochovice (3t)	33. Boskovice (1t)
7. Frýdek (11t)	16. Jeseník (4t)	25. Lhota (3t)	34. Žatčany (1t)
8. Petřvald (11t)	17. Šlapanice (4t)	26. Dolní Dunajovice (3t)	
9. Popice (10t)	18. Otnice (4t)	27. Střítež (3t)	

### 1. fáze – vícekritériální rozhodování

Vícekritériální rozhodování bylo použito za účelem stanovení vah vrcholů grafu (dopravní sítě) pro lokační analýzu. Váhy vrcholů jsou závislé na více než jednom kritériu a jsou vyjádřeny pomocí váženého součtu hodnot kritérií.

Použitá kritéria<sup>30</sup> jsou následovná:

- K1 – množství materiálu vezeného do vrcholu [kg],
- K2 – počet objednávek [-],
- K3 – cena za skladování [Kč/den],
- K4 – vzdálenost vrcholu od Prahy [km].

Ke stanovení vah kritérií, bylo použito dvou metod. První metoda byla Fullerova metoda a druhá metoda pořadí. Vyřešená tabulka Fullerovy metody je uvedena v příloze 3 diplomové práce.

Tab. 8: Porovnání metod stanovení vah kritérií [autor]

	Pořadí	Fullerova	Rozdíl
K1	0,4	0,5	0,10
K2	0,3	0,33	0,03
K3	0,15	0,08	-0,07
K4	0,15	0,8	0,65

Srovnáním výpočtů obou metod bylo potvrzeno, že Fullerova metoda zvýhodňuje nejdůležitější kritérium oproti ostatním více než metoda pořadí. K výpočtu úlohy byla proto použita metoda pořadí, aby byla kritéria vzájemně více rovnocenná.

<sup>30</sup> Seřazeno od nejdůležitějšího po nejméně důležité kritérium.

Pro efektivní výpočet bylo potřeba vytvořit krátký program v MATLABu. Výstupem programu je vektor váženého součtu hodnot kritérií. Originální zdrojový kód MATLABu je v příloze 4.

### Pseudokód programu na výpočet VHV

Načtení a deklaráce proměnných

*načtení tabulku MS Excel do matice A (jednotlivé hodnoty kritérií zapsány ve sloupcích)*

*deklaráce proměnné c a w (c = počet sloupců matice A, w = počet řádků matice A)*

*for cyklus: i=1 až c*

*konzolové zadání hodnot pořadové funkce  $b_k$  do sloupcového vektoru k*

*konzolové zadávání typu stupnice (1=poměrová, 0=pořadová) do sloupcového vektoru*

*k2*

*pokud k2== 0*

*je aktuální prvek sloupcového vektoru k1== 1*

*a pokračuje nové kolo for cyklu*

*konec podmínky if*

*konec for cyklu*

Metoda pořadí

*for cyklus: i=1 až c*

*zapsání hodnoty pořadové funkce do sloupcového vektoru b pomocí vzorce (36)*

*konec for cyklu*

*součet prvků vektoru b (kontrola správnosti výpočtu)*

Převod kritérií z minimalizačních na maximalizační

*for cyklus: i=1 až w*

*for cyklus: j=1 až c*

*proměnná D se rovná maximálnímu prvku j-tého sloupce matice A*

*proměnná D1 se rovná minimálnímu prvku j-tého sloupce matice A*

*pokud se aktuální hodnota k1==1*

*aktuální prvek matice Q se rovná prvku matice A se stejnými*

*indexy*

*jinak aktuální prvek matice Q je vypočten podle vzorce (29)*

*konec podmínky if*

*konec for cyklu*

*konec for cyklu*

Normované váhy kritérií

*for cyklus: j=1 až c*

*for cyklus: i=1 až w*

*proměnná D se rovná maximálnímu prvku j-tého sloupce matice Q*

*aktuální prvek matice Q1 se vypočte podle vzorce (32)*

*konec for cyklu*

*konec for cyklu*

Vážené normované hodnoty

*for cyklus: j=1 až c*

*for cyklus: i=1 až w*

*aktuální prvek matice Q2 se vypočte dle vzorce  $v_j * f_{i,j}$*

*konec for cyklu*

*konec for cyklu*

Součet řádků

*for cyklus: i=1 až w*

*aktuální prvek sloupcového vektoru Q3 se vypočte součtem řádku*

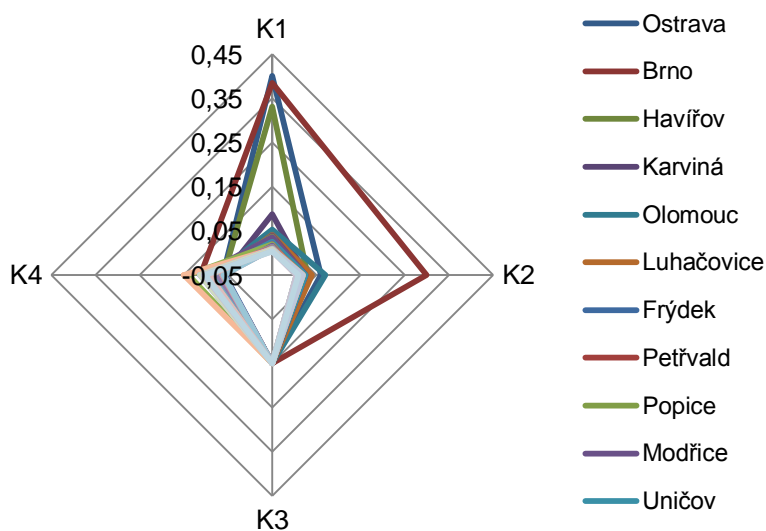
*konec for cyklus*

*zápis vektoru Q3 do MS Excelu (výstup programu)*

Výstupem programu je vektor váženého součtu hodnot kritérií. Podle indexu  $j$  lze každou hodnotu přiřadit k danému vrcholu. Číslování odpovídá 0. fázi, ve které bylo provedeno seřazení měst. Hodnoty výsledného vektoru (v programu značen Q3) jsou následovné:

$Q3=(0,672; 0,946; 0,571; 0,321; 0,353; 0,303; 0,277; 0,274; 0,297; 0,320; 0,298; 0,240; 0,234; 0,319; 0,251; 0,276; 0,290; 0,270; 0,290; 0,254; 0,247; 0,253; 0,273; 0,280; 0,237; 0,273; 0,228; 0,233; 0,323; 0,291; 0,254; 0,274; 0,264; 0,266)$

V grafu 4 je grafické znázornění úlohy VHV. Jednotlivé osy představují čtyři použité kritéria. Kvůli přehlednosti je v grafu zobrazeno pouze deset nejdůležitějších variant (měst). Další varianty mají velmi podobné hodnoty kritérií a v grafu se překrývají.



Graf 4: Grafické znázornění úlohy VHV [autor]

Metodou globálního kritéria bylo určeno pořadí jednotlivých variant. Metoda byla popsána v kapitole 3.3.5 a k nalezení řešení používá maximalizaci váženého součtu hodnot kritérií. Tři nejlepší varianty<sup>31</sup> jsou Brno, Ostrava a Havířov.

1. Brno – vážený součet hodnot kritérií: 0,946
2. Ostrava - vážený součet hodnot kritérií: 0,672
3. Havířov - vážený součet hodnot kritérií: 0,571

<sup>31</sup> Dopotčítání VHV neslouží k řešení úlohy. Bylo provedeno pro názornost, které vrcholy mají nejvyšší váhu.

## 2. fáze – lokační analýza

Před výpočtem úlohy lokační analýzy je zapotřebí sestavit dopravní síť. Množina vrcholů  $V$  jsou obce uvedené ve fázi 0. Ke stanovení správných ohodnocení hran (vzdáleností mezi vrcholy), byl použit freeware MapWindow do kterého byl nahrán soubor s příponou *gis* obsahující obce a komunikace celé České republiky.

Program pro výpočet lokační analýzy je stejně jako program předchozího kroku napsán v MATLABu a umožňuje umístění depa do libovolného vrcholu sítě. Obsahuje navíc možnost zvolení množiny kandidátů, čímž lze redukovat množství procházených kombinací při výpočtu. Program je napsán tak, aby uživatele vedl. Přesto práce s programem vyžaduje základní znalost lokační analýzy a softwaru MATLAB.

Program řeší lokační úlohu pomocí iterativního algoritmu nebo exaktní metody. Iterativní algoritmus byl schopen určit lokaci 33 dep, tedy nejvyššího možného počtu použité dopravní sítě. Doba výpočtů pomocí iterativního algoritmu roste lineárně a nedostala se nad jednu sekundu. Exaktní metodu hrubé síly bylo možné použít pro lokaci maximálně šesti dep (344 904 kombinací, doba výpočtu 150 sekund). Vyšší počet má za následek neúnosnou dobu výpočtu. Při lokaci sedmi dep algoritmus hrubé síly propočítává až 5 379 616 kombinací. Doba výpočtu exaktní metody roste exponenciální řadou.

### Pseudokód a popis práce s programem

#### Zvolení metody výpočtu

Použitou metodu volí uživatel po dotazu pomocí konzole a to následovně:

- Vložení čísla 1 – exaktní metoda hrubé síly,
- vložení čísla 2 – iterativní algoritmus.

*konzolové zadání proměnné iterativhrub (1 nebo 2)*

*switch iterativhrub*

*case 1 (část zdrojového kódu s algoritmem hrubé síly)*

*case 2 (část zdrojového kódu s iterativním algoritmem)*

*konec switch*

#### Zadání dopravní sítě do programu

Možnosti zadání dopravní sítě do programu jsou následovně:

- Postupný zápis do konzole – vložení čísla vrcholu (předchůdce), vložení čísla vrcholu (následovníka), vložení ohodnocení hrany  $o(h)$ ; program se ptá, dokud nejsou všechny vrcholy a hrany zadány,



- nahrání distanční matice – nahrání distanční matice pomocí textového souboru s příponou *txt*,
- nahrání grafu – nahrání grafu do souboru MS Excel; První sloupec představuje předchůdce, druhý sloupec následovníky a třetí sloupec příslušné ohodnocení hran.

U druhého a třetího způsobu nahrání je potřeba v programu zapsat správnou cestu k souboru. To samé platí i u nahrání vektoru  $Q_3$  z 1. fáze výpočtu.

*konzolové zadání proměnné v (1 nebo 0)*

*switch v*

*case 1*

*načtení distanční matice z textového souboru*

*nebo načtení se souboru MS Excel*

*vytvoření grafu G, vypočítání distanční matice  $d=\text{distances}(G)$*

*case 0*

*konzolové zadání proměnné x (x=počet hran grafu)*

*konzolové zadání předchůdce, následovníka a incidující hrany*

*vytvoření grafu G, vypočítání distanční matice  $d=\text{distances}(G)$*

*konec switch*

### **Zvolení počtu umíst'ovaných dep**

Po zadání dopravní sítě se program zeptá, kolik chce uživatel umístit dep. Počet se jednoduše zapíše do konzole. Jestliže uživatel zadá počet špatně (například číslo 0, nebo číslo vyšší než je počet vrcholů dopravní sítě) je program ukončen.

*konzolové zadání proměnné k (k=počet dep)*

### **Zadání množiny kandidátů**

Množina kandidátů se do programu zadává pomocí konzole.

*konzolové zadání proměnné z ( $z=(0, 1)$ )*

*pokud  $z=1$*

*konzolové zadání proměnné z1 ( $z1=\text{počet vrcholů množiny kandidátů}$ )*

*for cyklus:  $i=1$  až  $z1$*

*konzolové zadání vektoru  $z(1,i)$*

*konec for cyklu*

*jinak vektor  $z=1$  až  $y$  ( $y=\text{všechny vrcholy grafu}$ )*

*konec podmínky*

## **Uložení výstupu**

Uložení výstupu je v MATLABu řešeno funkcí *diary*. Funkce ukládá do textového souboru vše, co se během výpočtu vypíše do konzole. Uložení výsledného grafu musí uživatel provést sám pomocí exportu.

*konzolové zadání proměnné vystup (vystup 0,1)*

```
switch vystup
    case 1
        ukládání konzole zapnuto
    case 0
konec switch
```

*Na konci zdrojového kódu*

```
switch vystup
    case 1
        uložení textového souboru s příponou txt
    case 0
konec switch
```

## **Pseudokód algoritmu hrubé síly**

Část programu lokační analýzy řešící algoritmus hrubé síly je rozdělený na dvě části. První část počítá dopravní práci počátečního řešení. Počáteční řešení je pole (vektor) s prvky od 1 do  $k$ . Druhá část prochází všechny kombinace vrcholů a počítá příslušnou dopravní práci. Dopravní práce jednotlivých řešení se v průběhu algoritmu porovnávají. Program zapisuje nejlepší nalezené řešení s příslušnou hodnotou dopravní práce.

Nekonečný while cyklus (while cyklus s podmínkou  $1 > 0$ ) je jak u první tak u druhé části zdrojového kódu programu použit kvůli funkcím *break* a *continue*. V druhé části navíc while cyklus slouží ke generování všech možných kombinací vrcholů.

Výpočet dopravní práce pro počáteční řešení 1 až  $k$

*n=počet vrcholů grafu*

*while cyklus (s nekonečně trvajícím podmínkou  $1 > 0$ )*

*funkce ismember porovnává vektor pole a vektor z*

*pokud pole obsahuje jiný prvek než prvky z vektoru z – break (program opustí While cyklus)*

*for cyklus:  $i=1$  až  $k$*

*vektor b = i-tý řádek distanční matice d %vytvoření matice b*

*konec for cyklu*

```

for cyklus: j=1 až n
    proměnná v=min prvek j-tého sloupce matice b
    když v=0 continue
     $D=(2*v*váha\ j-tého\ vrcholu)$ 
    a zapíšeme si hodnotu do vektoru a
konec for cyklu
D111=součet prvků vektoru a
proměnná pole1 = pole
break
konec podmínky while
D11=D111 a vynulování vektoru a

```

Výpočet dopravní práce pro všechny ostatní možné řešení

$n$ =počet vrcholů grafu,  $n1=n$ ;

while cyklus (s nekonečně trvajícím podmínkou  $1>0$ )

```

for cyklus: i=k až 1, (krok=-1)
    pokud i-tý prvek pole <  $n-(k-i)$ 
        i-tý prvek pole=i-tý prvek pole+1;
        for cyklus: j=i+1 až k
            j-tý prvek pole=j-tý-1 prvek+1
        konec for cyklu
    break;
konec podmínky if
konec for cyklu

```

funkce ismember porovnávající vektor pole a vektor z

pokud pole obsahuje jiný prvek než prvky z vektor z – continue (program přejde do nového cyklu While cyklu)

```

pokud používám množinu kandidátů ( $z11=1$ )
     $n1=délka\ vektoru\ z\ (počet\ kandidátů)$ 
for cyklus: i=1 až k
    vektor b = i-tý řádek distanční matice d %vytvoření matice b
konec for cyklu
for cyklus: j=1 až n ( $n=počet\ vrcholů\ grafu$ )
    proměnná v=min prvek j-tého sloupce matice b
    když v=0 continue
     $D=(2*v*váha\ j-tého\ vrcholu)$ 
    a zapíšeme si hodnotu do vektoru a
konec for cyklu
D1=součet prvků vektoru a

```

```

pokud D11>=D1
    D11=D1 a pole1=pole
konec podmínky if
vynulování vektoru a
    pokud první prvek pole == n1-k+1
        break
    konec podmínky if

```

konec while cyklu

### **Pseudokód iterativního algoritmu**

Část programu lokační analýzy, iterativní algoritmus, řeší pouze umístování 2 a více dep. Lokace jednoho depa je vždy řešena algoritmem hrubé síly. Program funguje jako upravený iterativní algoritmus popsáný v kapitole 3.2.2.

for cyklus:  $i=1$  až  $k$

    konzolové zadávání prvků vektoru počáteční množiny  $a1$

konec for cyklu

while cyklus  $1>0$

    vektor pole = vektor počáteční množiny

    vektor  $b1$ =prvky 1 a  $n$  (obsahuje všechny vrcholy)

    vektor pole1= $a1$  (současné nejlepší řešení je počáteční množina)

    pokud  $z11==1$

$B1=z$

$N1$ =počtu prvků vektoru  $b1$

    konec podmínky if

vektor proměnných  $c1$  vytvořený funkcí `ismember(b1,a1)`

for cyklus:  $i=1$  až  $n1$

    pokud  $i$ -tý prvek vektoru  $c1==0$

        zapiše se do vektoru `vysledek`  $i$ -tý prvek z vektoru  $b1$

    jinak `continue`

    konec podmínky if

konec for cyklu

oříznutím nuly z vektoru `vysledek` vytvoříme vektor `vysledek1`

$D11=D111$  %výsledek z počáteční množiny zapsán do proměnné  $D11$

for cyklus:  $x=1$  až  $(n1-k)$  ( $n1=n$ , nerovnost pouze v případě práce s množinou kandidátů)

    for cyklus:  $j=1$  až  $k$

        pomocný vektor  $a$  do kterého je zapsáno  $a1$

        pomocná proměnná  $D3$ , do které je zapsána hodnota  $D11$

$j$ -tý prvek pole =  $x$ -tý prvek `vysledek1`

        for cyklus:  $i=1$  až  $k$

```

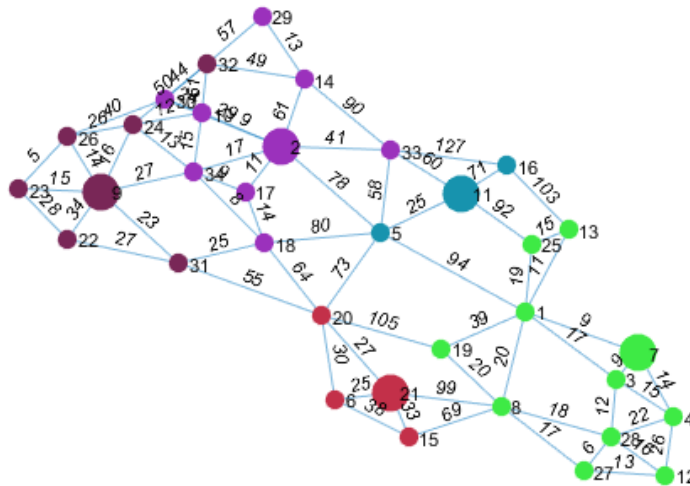
vektor b = i-tý řádek distanční matice d %vytvoření matice b
konec for cyklu
for cyklus: y=1 až n (n=počet vrcholů grafu)
    proměnná v=min prvek j-tého sloupce matice b
    pokud v=0 continue
    konec podmínky if
        D=(2*v*váha j-tého vrcholu)
        a zapíšeme si hodnotu do vektoru g
    konec for cyklu
        D1=sum(g);
        zápis D1 do pomocného vektoru D2
        pole = vektoru a
        vynulování pomocného vektoru g
    konec for cyklu
vektor D22 vytvořen z D2 oříznutím nuly na místě (1,1)
použití funkce find (Nalezneme minimální prvek vektoru D22 a jeho pořadí ve vektoru c)
    pokud D11>=D3
        Z1=z1+1; pomocná proměnná z iterativního algoritmu
        D11=D3
        c-tý prvek pole= x-tému prvku vektoru vysledek1
        vektor a= vektor pole (přepsání nového řešení)
        pole1=a %zapsání si výsledku stranou
    konec podmínky if
vynulování vektoru D2 a proměnné D3
konec for cyklu %zakončený průchod všech řešení (pole1 obsahuje nejlepší řešení)
pokud z1>0
    A1=pole1 (přepsání současného řešení)
jinak break (vyskočení z while cyklu konec iterativního algoritmu)
konec podmínky if
konec while cyklu

```

### Grafický výstup programu

Program MATLAB umožňuje funkcí *plot* vykreslit dopravní síť. Barva je náhodně generována pomocí for cyklu. Obrázek č. 7 představuje výstup programu lokační analýzy<sup>32</sup>. Vrcholy, které jsou zobrazeny větší než ostatní (pomocí dvojnásobné funkce *highlight*), představují vrcholy ze (sub)optimálního řešení  $D_k$ . Každý vrchol z množiny  $D_k$  má kolem sebe vrcholy stejné barvy, které tvoří přidělený atrakční obvod. Určení přidělených atrakčních obvodů je součástí zdrojového kódu v příloze 5.

<sup>32</sup> Jako příklad je uvedena lokace pěti dep.

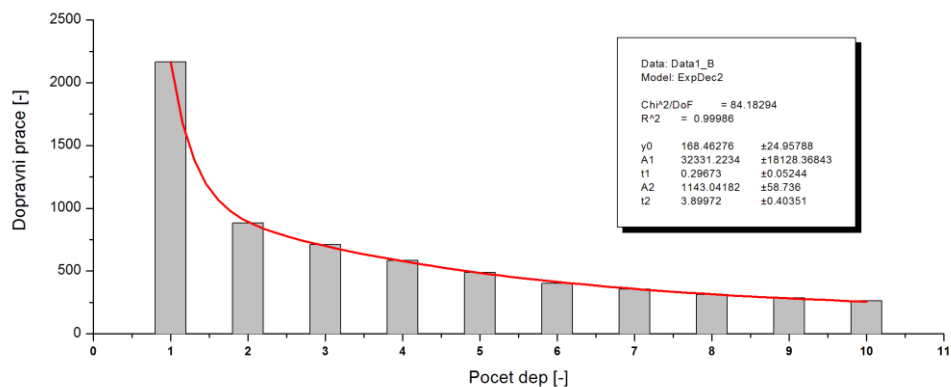


Obr. 7: Výstup programu [autor]

## Výsledky 2. fáze

Výsledky lokační analýzy jsou shrnuty v tabulce 9. Iterativní algoritmus je při lokaci 1 – 6 dep schopen řešit úlohu se shodnými výsledky jako algoritmus hrubé síly. Zároveň je čas výpočtu mnohonásobně nižší než u exaktní metody.

Provedením regresní analýzy ve statistickém programu STATGRAPHICS bylo dokázáno, že dopravní práce (hodnota účelové funkce) klesá s počtem dep exponenciálně.



Graf 5.: Exponenciální pokles dopravní práce v závislosti na počtu dep [autor]

Stejný průběh lze předpokládat i u nákladů za distribuci zboží ze skladů k zákazníkům. Úspory za dopravu se budou každým navýšením počtu skladů snižovat a naopak budou narůstat náklady za pronájem skladových prostor, manipulaci a dopravy na sklad. Proto je v dalších krocích počítáno pouze se zásobováním zákazníků z jednoho, dvou a tří skladů.

Tab. 9: Porovnání výsledků použitých metod [autor]

Počet dep	Algoritmus hrubé síly			Iterativní algoritmus		
		Čas výpočtu [s]	Dopravní práce [-]		Čas výpočtu [s]	Dopravní práce [-]
1	2	0,31593	2165,200	2	0,31593	2165,200
2	1, 2	0,40153	883,126	1, 2	0,322952	883,126
3	1, 2, 21	0,96494	709,272	1, 2, 21	0,374915	709,272
4	7, 10, 11, 21	5,44969	584,520	7, 10, 11, 21	0,392736	584,520
5	7, 10, 11, 14, 21	31,53323	488,060	7, 10, 11, 14, 21	0,434408	488,060
6	2, 7, 9, 11, 14, 21	150,27820	402,206	2, 7, 9, 11, 14, 21	0,480870	402,206

**Lokace jednoho depa:** Vrchol 2 – Brno – Štýřice

Průměrná vzdálenost k zákazníkům v Brně ze skladu Brno-Štýřice byla v roce 2014 6,2 km.

**Lokace dvou dep:** Vrcholy 1 a 2 – Brno – Štýřice, Ostrava – Poruba

Průměrná vzdálenost k zákazníkům v Ostravě ze skladu Ostrava - Poruba byla v roce 2014 6,05 km.

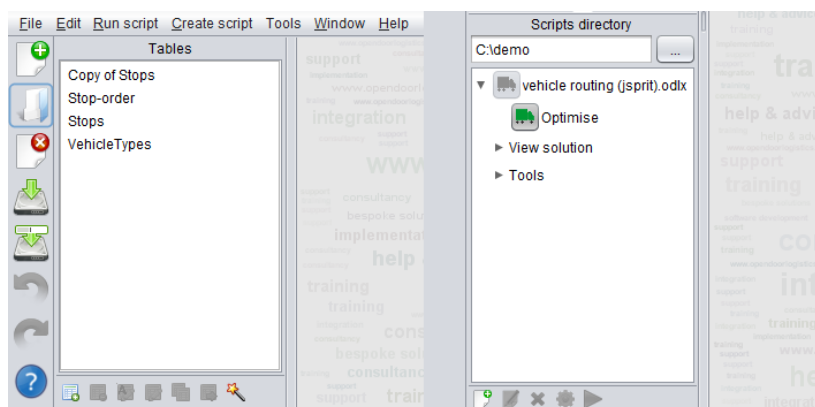
**Lokace tří dep:** Vrcholy 1, 2 a 21 – Brno – Štýřice, Ostrava – Poruba a Zlín

Průměrná vzdálenost k zákazníkům ve Zlíně z vybraného skladu byla v roce 2014 1,5 km.

Všechny vybrané sklad nabízí uskladnění 1-50 000 palet a služby spojené se skladováním.

### 3. fáze – okružní jízdy s časovými okny (VRPTW)

Ve třetí fázi je řešen výpočet okružních jízd. Přesněji jde o úlohu Vehicle Routing Problem with Time Windows (VRPTW), tedy úlohu okružních jízd s časovými okny. Časová okna definují čas, ve kterém je možné daný vrchol obsloužit (navštívit). K výpočtu úlohy bylo použito freewaru Open Door Logistics Studio. Program je jednoduchý na ovládání a je napsán v programovacím jazyku Java (viz. kapitola 3.2.3). Na obr. 8 je zobrazeno hlavní menu programu.



Obr. 8: Hlavní menu programu Open Door Logistics Studio [autor]

Program je určen především pro Německo, Švýcarsko, Velkou Británii, Itálii a další velké evropské země. Chce-li uživatel použít program pro Českou republiku, musí postupovat následovně:

1. Stažení souboru s příponou *pbf* z [graphhopper.com](http://graphhopper.com) ([czech-republic-latest.osm.pbf](http://czech-republic-latest.osm.pbf)),
2. vytvoření adresáře s názvem *temp* na disku *C* se souborem s příponou *pbf*,
3. otevření příkazového okna pomocí příkazu *cmd*,
4. vložení příkazu do příkazového okna ve tvaru

```
cd c:\Program Files\ODLStudio – 1.3.2 – Win – 64 – bit\graphhopper a
java –Xmx2G –jargraphhopper – 0.4 – SNAPSHOT – jar – with –
dependencies.jar config = config.properties osmreader.osm = c:\temp\czech –
republic – latest.osm.pbf.
```

Jestliže po provedení doporučeného postupu nebude v programu načtená Česká republika, je zapotřebí do adresáře *temp* stáhnout z webu [graphhopper.com](http://graphhopper.com) zazipovanou složku *czech-republic-latest.osm-gh*. Složka obsahuje soubory *hrany*, *uzly*, *lokace*, *geometrie*, *vlastnosti* a *jména*. V nastavení optimalizačního okna pro okružní jízdy zvolíme cestu k rozbalené složce *czech-republic-latest.osm-gh*.

Před výpočtem úlohy okružních jízd musí uživatel zadat *Stops* (zastávky) a *VehicleTypes* (typy použitých automobilů) do tabulek v pravé horní části menu programu.

Tabulka zastávek se vytváří jednoduchým vložením názvů všech navštěvovaných obcí. U obcí se vždy definuje doba potřebná k vyložení nákladu a časové okno. Čas potřebný k vyložení nákladu byl stanoven na 30 minut. Časové okno bylo nastaveno od 8:00 do 17:00. Po vyplnění tabulky je třeba kliknout na ikonu „kouzelné hůlky“ a zvolit *Geocode addresses using Nominatim*. Jedná se o zadávání souřadnic měst pomocí map z webové stránky [openstreetmap.org](http://openstreetmap.org).

Tabulka typů automobilů se vytvoří postupným zadáním všech parametrů vozidel. Každý typ vozidla musí mít zadané své depo, z kterého automobil vyjíždí a kam se po obslužení zákazníků vrací. Dalším parametrem je kapacita vozidla, cena za kilometr a jeho množství ve vozovém parku. Tabulka 10 byla vytvořena podle tabulky cen za km (tabulka 3).



Tab. 10: Tabulka typů vozidel [autor]

Název	Start - z.š. <sup>33</sup>	Start - z.d.	Konec -z.š.	Konec - z.d.	Kapacita	Cena za km [Kč]
D1	49,192244	16,6113382	49,192244	16,6113382	3000	14
D2	49,8349139	18,2820084	49,8349139	18,2820084	3000	14
D3	49,192244	16,6113382	49,192244	16,6113382	6000	16,5
D4	49,8349139	18,2820084	49,8349139	18,2820084	6000	16,5
D5	49,192244	16,6113382	49,192244	16,6113382	9000	19
D6	49,8349139	18,2820084	49,8349139	18,2820084	9000	19
D7	49,192244	16,6113382	49,192244	16,6113382	1600	10
D8	49,8349139	18,2820084	49,8349139	18,2820084	1600	10

Po zadání tabulky zastávek a automobilů lze spustit optimalizaci s nastaveným počtem iterací. Počet iterací má vliv na přesnost výsledků. Po výpočtech si lze v levém dolním rohu programu zobrazit výsledky, které umožňují:

- Editaci tras,
- zobrazení trasy v mapě
- zobrazení detailu zastávek, tras, řešení (počet použitých tras, doručené množství, čas trvání rozvozu, cena, ujeté kilometry)
- souhrnný výstup optimalizace ve formátu *pdf*,
- Gantův diagram,
- graf nákladů,
- export dat.

### Výsledky 3. fáze

K určení nákladů na dopravu k zákazníkům z nových skladů v roce 2014 bylo zapotřebí vyřešit okružní jízdy pro každé pondělí a středu (dny závozu na Moravu) tohoto roku. Program dokázal vypočítat ceny za dopravu. Výpočty byly rozděleny do čtyř období definovaných v kapitole 2.5.

Problém výpočtů nastává při nutnosti rozvozu zboží ve městě, ve kterém se nachází sklad. Program určí nulovou vzdálenost a tedy i nulové náklady na dopravu. V takovémto případě byla použita průměrná vzdálenost ze skladu k zákazníkům v daném městě. Hodnoty jsou uvedeny v předchozím kroku lokační analýzy. Průměrná vzdálenost byla násobena příslušnou hodnotou z tabulky cen za km podle velikosti dodávky. Vykalkulované náklady na dopravu v roce 2014 z lokační analýzy umístěných skladů jsou zapsány v tabulce 11.

<sup>33</sup> Z.š. – zeměpisná šířka, z.d. – zeměpisná délka.

Tab. 11: Vypočtené náklady na dopravu k zákazníkům [autor]

Počet dep	Náklady na dopravu k zákazníkům [Kč]				Celkem [Kč]
	Útlum 1	Špička 1	Útlum 2	Špička 2	
1	46433	164050	68073	111882	390 438
2	19742	46392	30707	46304	143 146
3	15943	36808	22923	36312	111 985

#### 4. fáze – stochastický model zásob

Vypočtenou tabulku 11 z předchozího kroku nelze použít k porovnání s náklady na dopravu společnosti Sto, s.r.o. za rok 2014. K hodnotám nákladů na dopravu k zákazníkům je třeba přičíst náklady na skladování. Protože není předem známá velikost spotřeby, byl k nalezení hodnot  $m_0$  (optimální velikost dodávky) a  $h_0$  (optimální hladina objednáni) použit stochastický model se signalizací změn a ztracenými prodeji. Jak bylo popsáno v teoretické části, ztracený prodej znamená, že vzniklý deficit není nahrazován z následující dodávky. V řešeném problému je očekáván velký interval mezi jednotlivými dodávkami, proto nahrazování zboží z následujících dodávek není možné. Čas hraje ve stavebním průmyslu důležitou roli. Z vypočítaných hodnot  $m_0$  a  $h_0$  lze vzorcem (59) dopočítat průměrný počet palet na skladě  $\bar{m}$  a vzorcem (60) střední počet dodávkových cyklů  $n$ . Tyto hodnoty jsou důležité pro stanovení nákladů na skladování.

Výpočet byl opět proveden pomocí jednoduchého programu napsaného v MATLABu (příloha 7). Program vychází z postupu a vzorců z kapitoly 3.4. Dopočítávání iterací a zpřesňování hodnot  $m$  a  $h$  je řešeno pomocí for cyklu, který je explicitně nastaven na 10 opakování. Před spuštěním programu je zapotřebí spočítat střední hodnotu deficitu  $EU$  pro rovnoměrné rozdělení spotřeby. Výsledný  $EU$  se zapisuje přímo do zdrojového kódu programu. Model byl řešen pro každý typ malty, počet skladů a období<sup>34</sup> zvlášť. Zvolenou jednotkou výpočtu je paleta<sup>35</sup>.

#### Výsledky 4. fáze

Výsledky jsou shrnuty v tabulce 12. K výpočtu ceny za skladování je třeba znát průměrný počet palet na skladě  $\bar{m}$  a střední počet dodávkových cyklů  $n$ . Tyto dvě veličiny jsou zvýrazněny červeně. Tabulka obsahující výpočty pro jednotlivé druhy malt je uvedena v příloze 6. Podle zástupce webu *skladuj.cz* je průměrná cena za skladování jedné palety 3,75 Kč/den a manipulace s paletou stojí průměrně 30 Kč (IN i OUT).

<sup>34</sup>Útlum 1, špička 1, útlum 2 a špička 3.

<sup>35</sup> Uváděné ceny poskytovateli skladovacích služeb jsou vztaženy často na jednotku paleta.

Tab. 12: Výsledky stochastického modelu zásob [autor]

Počet dep - k	Útlum 1			Špička 1			Útlum 2			Špička 2		
	1	2	3	1	2	3	1	2	3	1	2	3
Celkem $m_0$ [paleta]	86	252	477	285	592	1035	181	460	747	175	528	1143
Celkem $h_0$ [paleta]	17	30	36	40	72	102	47	94	135	36	62	87
Celkem $\bar{m}$ [paleta]	<b>51</b>	<b>74</b>	<b>96</b>	<b>84</b>	<b>114</b>	<b>195</b>	<b>88</b>	<b>152</b>	<b>207</b>	<b>75</b>	<b>118</b>	<b>165</b>
Celkem $n$ [-]	<b>4</b>	<b>8</b>	<b>12</b>	<b>11</b>	<b>12</b>	<b>15</b>	<b>7</b>	<b>10</b>	<b>12</b>	<b>7</b>	<b>12</b>	<b>18</b>

K ceně za manipulaci s paletami a za pronájem paletových míst byla připočítaná cena za dopravu na sklad. Současný dopravce Trade trans Log s.r.o. přepravující zboží z Německa na sklad v Dobřejovicích, nabízí přepravu za cenu 33 Kč/km. Protože cílem kapitoly je určit náklady vyplývající z využívání nového řešení, byla cena za dopravu na sklad obsažena v nákladech na skladování. Výpočty nákladů na skladování v jednotlivých obdobích byly provedeny podle vzorce (61). Vzorec představuje součet nákladů na pronájem paletových míst, nákladů na dopravu na sklad a nákladů na manipulaci s paletami.

$$N_{k;období} = (\bar{m} * 3,75 * \text{počet dnů období}) + (n * \text{cena dopravy na sklad}) + (m * 2 * 30), \quad (61)$$

kde index  $k$  značí počet skladů.

### Výpočet nákladů na skladování ( $k = 1$ )

$$N_{1;útlum\ 1} = (51 * 3,75 * 120) + (4 * 6500^{36}) + (86^{37} * 2 * 30) = 54110 \text{ Kč}$$

$$N_{1;špička\ 1} = (84 * 3,75 * 90) + (11 * 6500) + (285 * 2 * 30) = 116950 \text{ Kč}$$

$$N_{1;útlum\ 2} = (88 * 3,75 * 60) + (7 * 6500) + (181 * 2 * 30) = 76160 \text{ Kč}$$

$$N_{1;špička\ 2} = (75 * 3,75 * 90) + (7 * 6500) + (175 * 2 * 30) = 81313 \text{ Kč}$$

$$N_1 = 328\ 533 \text{ Kč}$$

### Výpočet nákladů na skladování ( $k = 2$ )

$$N_{2;útlum\ 1} = (74 * 3,75 * 120) + (4 * 6500) + (4 * 12000^{38}) + (252 * 2 * 30) = 122420 \text{ Kč}$$

Sloučením dodávek došlo k finanční úspoře.

$$N_{2;útlum\ 1} = 122420 - (12000 + 6500) = 103920 \text{ Kč}$$

$$N_{2;špička\ 1} = (114 * 3,75 * 90) + (6 * 6500) + (6 * 12000) + (592 * 2 * 30) = 184995 \text{ Kč}$$

$$N_{2;útlum\ 2} = (152 * 3,75 * 60) + (5 * 6500) + (5 * 12000) + (460 * 2 * 30) = 154300 \text{ Kč}$$

$$N_{2;špička\ 2} = (118 * 3,75 * 90) + (6 * 6500) + (6 * 12000) + (528 * 2 * 30) = 182505 \text{ Kč}$$

$$N_2 = 625\ 720 \text{ Kč}$$

<sup>36</sup> Cena za dopravu Praha – Brno = 6500 Kč.

<sup>37</sup> Počet poslaných palet.

<sup>38</sup> Cena za dopravu Praha – Ostrava = 12000 Kč, Praha – Zlín = 9750 Kč.

### Výpočet nákladů na skladování ( $k = 3$ )

$$N_{3;\text{útlum } 1} = (96 * 3,75 * 120) + (4 * 6500) + (4 * 9750) + (4 * 12000) + (477 * 2 * 30) = \\ = 184820 \text{ Kč}$$

Sloučením dodávek došlo k finanční úspoře.

$$N_{3;\text{útlum } 1} = 184820 - (12000 + 9750 + 6500) = 156570 \text{ Kč}$$

$$N_{3;\text{špička } 1} = (195 * 3,75 * 90) + (5 * 6500) + (5 * 9750) + (5 * 10000) + (1035 * 2 * 30) = \\ = 259163 \text{ Kč}$$

$$N_{3;\text{útlum } 2} = (207 * 3,75 * 60) + (4 * 6500) + (4 * 9750) + (4 * 12000) + (747 * 2 * 30) = \\ = 204395 \text{ Kč}$$

$$N_{3;\text{špička } 2} = (165 * 3,75 * 90) + (6 * 6500) + (6 * 9750) + (6 * 12000) + (1143 * 2 * 30) = \\ = 293768 \text{ Kč}$$

Sloučením dodávek došlo k finanční úspoře.

$$N_{3;\text{špička } 2} = 293768 - (12000 + 9750 + 6500) = 265518 \text{ Kč}$$

$$N_3 = 1\,179\,414 \text{ Kč}$$

### 5. fáze – ekonomické zhodnocení

Celkové náklady na dopravu na Moravu a do Slezska společnosti Sto, s.r.o. v roce 2014 byly 1 656 733 Kč. Za distribuci Malt na Moravu a do Slezska zaplatila společnost 823 155 Kč. Celkové náklady nových řešení se stanoví jako součet nákladů na skladování a nákladů na dopravu k zákazníkům. Náklady na dopravu k zákazníkům a skladování jsou zapsány v tabulce 13.

Tab. 13: Vypočtené náklady na dopravu k zákazníkům a na skladování [autor]

Počet dep	Cena za dopravu [Kč]				Celkem [Kč/rok]
	Útlum 1	Špička 1	Útlum 2	Špička 2	
1	46433	164050	68073	111882	390438
2	19742	46392	30707	46304	143146
3	15943	36808	22923	36312	111985
Počet dep	Cena za skladování [Kč]				Celkem [Kč/rok]
	Útlum 1	Špička 1	Útlum 2	Špička 2	
1	54110	116950	76160	81313	328533
2	103920	184995	154300	182505	625720
3	156570	259163	204395	469912	1179414

### Celkové roční náklady pro řešení s jedním skladem v Brně

$$N_{1\text{celkem}} = 390438 + 328533 = 718\,971 \text{ Kč}$$

$$N_{1\text{úspora}} = 823155 - 718971 = 104\,184 \text{ Kč (13 \%)}$$

### Celkové roční náklady pro řešení se dvěma sklady v Brně a Ostravě

$$N_{2celkem} = 143146 + 625720 = 768866 \text{ Kč}$$

$$N_{1úspora} = 823155 - 728\,866 = 54\,289 \text{ Kč (7 \%)}$$

### Celkové roční náklady pro řešení se třemi sklady v Brně, Ostravě a Zlíně

$$N_{3celkem} = 111985 + 1179414 = 1\,291\,399 \text{ Kč}$$

$$N_{3úspora} = 823155 - 1\,291\,399 = -468\,244 \text{ Kč (-57 \%)}$$

Úspora při řešení distribuce na Moravu a do Slezska pronajmutím jednoho skladu je 104 183 Kč/rok (13 %). Výše úspory byla negativně ovlivněna obdobím špička 1. Toto období bylo specifické. Prodalo se zde největší množství zboží při nejmenším počtu objednávek. Z toho vyplývá, že dovoz byl realizován po velkém množství v jedné zásilce. Zároveň se v tomto období náhodně sešlo velké množství materiálu vezeného do Ostravy, Karviné a okolí. Proto je v tabulce 13 patrný výrazný rozdíl mezi náklady řešení s jedním skladem a náklady řešení s více sklady. Cena za dopravu ze skladu v Brně v tomto období vychází 281 000 Kč, což je dokonce o 9 210 Kč více než současné řešení dovozu z Dobřejovic. Přitom při podobném množství prodaného materiálu v období špička 2 tvoří úspora až 57 072 Kč. Při rovnoměrnějším rozmístění míst vzniku poptávky a vyšším počtu objednávek, lze u řešení s jedním skladem očekávat vyšší finanční úsporu (cca 16 - 18 %). Řešením tohoto problému, může být neskladování materiálu Levell Duo. Tato malta je objednávaná po velkém množství a převážně do Ostravy a okolí. Řešení je blíže popsáno a kalkulováno v kapitole 5.

Varianta se dvěma sklady přináší ročně úsporu 54 288 Kč (7 %). Největší problém tohoto řešení jsou drahé dodávky na sklad v Ostravě. Přestože byl jejich počet optimalizován, stále tato položka tvoří podstatnou část nákladů. Náklady varianty se třemi sklady jsou vyšší než současné řešení distribuce z Dobřejovic a to o 468 244 Kč. Stejně jako v řešení se dvěma sklady i zde hraje velkou roli drahé zásobování skladů v Ostravě a ve Zlíně. Žádné ze třech zkoumaných řešení nepřineslo úsporu v zimních měsících (viz tabulka 14).

Tab. 14: Celkové náklady nových řešení [autor]

Počet skladů	Náklady nových řešení [Kč/rok]			
	Útlum 1	Špička 1	Útlum 2	Špička 2
1	100543	281000	144233	193195
2	123662	231387	185007	228809
3	172513	295971	227318	506224
<b>Současné řešení [Kč/rok]</b>	<b>98157</b>	<b>271790</b>	<b>202941</b>	<b>250267</b>

Možné zefektivnění nově nalezených řešení lze docílit použitím levnější dopravy z Německa na sklady, respektive dopravy z Dobřejovic na Moravu a do Slezska.

Současná doprava plného kamionu (26 palet) stojí z Tollwitz do Dobřejovic 8000 Kč. Vzdálenost obou měst je 300 km. Cena za jeden kilometr je 27 Kč. Cena okolo jednoho eura je v kamionové přepravě standardní. Dopravce Trade trans Log s.r.o. vozí materiál za tuto cenu pouze na trase Tollwitz – Dobřejovice. Od Prahy na Moravu a do Slezska vyžaduje za kilometr 33 Kč. Vyšší cenu dopravce stanovil z důvodu, že kamión nemá v Brně dohodnutou „proti-nakládku“. Nabízí se použití dopravce Zach Trans, s.r.o., který je ochoten přepravovat zboží kamionem na Moravu za 25 Kč/km. Řešení s použitím tohoto dopravce je uvedeno v následující kapitole.

## **5. Doporučené řešení**

Doporučit nové řešení není snadné, neboť rozhodnutí závisí na obchodní strategii společnosti Sto, s.r.o. Kdyby firma nadále zvyšovala své aktivity rovnoměrně po celé Moravě a Slezsku, dalo by se uvažovat o řešení s jedním skladem v Brně, později doplněném o sklad v Ostravě.

Společnost Sto, s.r.o. ale plánuje navýšit prodej na Jižní Moravě. Firma se chce začít soustředit mimo velkých panelových domů i na domy rodinné a ukončit obchodní aktivitu ve Slezsku a na Severní Moravě. Při takovémto strategickém plánu lze společnosti doporučit řešení s jedním skladem v Brně. Přerušením zásobování Ostravy a okolí, dojde k navýšení úspor razantním snížením najetých kilometrů při okružních jízdách. Navíc zásobování malých staveb bude vyžadovat vyšší frekvenci dodávek a flexibilnější reagování na objednávky. Sklad v Brně je pro takovéto zásobování ideálním řešením.

Dodávky na sklad jsou v doporučeném řešení prováděny dopravcem Zach Trans, s.r.o., který je schopen přepravovat zboží kamionem za 25 Kč/km viz tabulka 3. Materiál z Německého výrobního závodu v Tollwitz by byl dovážen společností Trade trans Log s.r.o. Po příjezdu do Dobřejovic dojde k překládce zboží na kamion společnosti Zach Trans, s.r.o. Přeložení zboží může být vykonáno i v jiný den vyskladněním zboží z hlavního skladu. Manipulace s materiálem by nebyla zpoplatněna, neboť by ji vykonávali zaměstnanci společnosti Sto, s.r.o. v rámci své pracovní doby a to pouze několikrát za měsíc, místo nakládání automobilů na Moravu dvakrát za týden. Zaměstnanci skladu by nebyli nadbytečně vytěžováni.

Další část doporučeného řešení je úplné vynechání malty Levell Duo. Tato malta je objednávaná po velkých množstvích, což zvyšuje náklady na pronájem paletových míst a manipulaci. Skladem by procházelo velké množství palet a bylo by zde potřeba držet zbytečně vysokou pojistnou zásobu. Velké zásilky se vyplatí posílat přímo z Dobřejovic. Navíc materiál Levell Duo byl poptáván pouze v Ostravě a Karviné. Tyto města se v budoucnu nebudou pravidelně zásobovat.

### **5.1 Výpočet nákladů doporučeného řešení**

K výpočtu nákladů je použit stejný postup jako při předcházející kalkulaci. Pomocí vzorce (61) stanovíme náklady na skladování v jednotlivých obdobích. Přičemž cena za kamionovou dopravu (Zach Trans, s.r.o.) z Dobřejovic do Brna je 5000 Kč (25 Kč/km).

$$N_{1,útlum\ 1} = (51 * 3,75 * 120) + (4 * 5000) + (86 * 2 * 30) = 47660\text{Kč}$$

$$N_{1,špička\ 1} = (84 * 3,75 * 90) + (11 * 5000) + (285 * 2 * 30) = 100450\text{Kč}$$

$$N_{1,útlum\ 2} = (88 * 3,75 * 60) + (7 * 5000) + (181 * 2 * 30) = 65660\text{Kč}$$

$$N_{1,špička\ 2} = (75 * 3,75 * 90) + (7 * 5000) + (175 * 2 * 30) = 70812\text{Kč}$$

$$N_1 = 284\ 582\text{Kč}$$

Celkové náklady  $N_{1celkem}$  řešení s jedním skladem v Brně se stanoví jako součet nákladů na skladování a nákladů na dopravu k zákazníkům.

$$N_{1celkem} = 390438 + 284582 = 675\ 020\text{Kč}$$

Dále je potřeba odečíst od nákladů na skladování<sup>39</sup> materiálu Levell Duo náklady na dopravu<sup>40</sup> tohoto materiálu z Dobřejovic na Moravu a do Slezska v roce 2014. Následným odečtením nákladů na skladování materiálu Levell Duo od celkových nákladů  $N_{1celkem}$  lze stanovit celkové náklady doporučeného (navrhovaného) řešení  $N$ .

$$N_{LevellDuo} = 80\ 070\text{Kč} - 53\ 119\text{Kč} = 26\ 951\text{Kč}$$

$$N = N_{1celkem} - N_{LevellDuo} = 648\ 069\text{Kč}$$

Výpočet velikosti úspory doporučeného řešení  $N_{úspora}$  se provede odečtením nákladů doporučeného řešení od nákladů společnosti Sto, s.r.o. na distribuci malt na Moravu a do Slezska v roce 2014.

$$N_{úspora} = 823\ 155\text{Kč} - 648\ 069\text{Kč} = 175\ 086\text{Kč} \text{ (21,3 \%)}$$

## 5.2 Shrnutí doporučeného řešení

Následující tabulka 15 shrnuje základní údaje doporučeného řešení.

Tab. 15: Shrnutí doporučeného řešení [autor]

Umístění skladu	Brno – Štýřice
Dopravce	Zach Trans, s.r.o.
Průměrný počet palet na skladě	51 palet/rok
Průměrný počet dodávek na sklad	20 dodávek/rok

<sup>39</sup> Vypočteno pomocí stochastického modelu zásob (viz příloha 7).

<sup>40</sup> Stanoveno dle dat společnosti Sto, s.r.o.



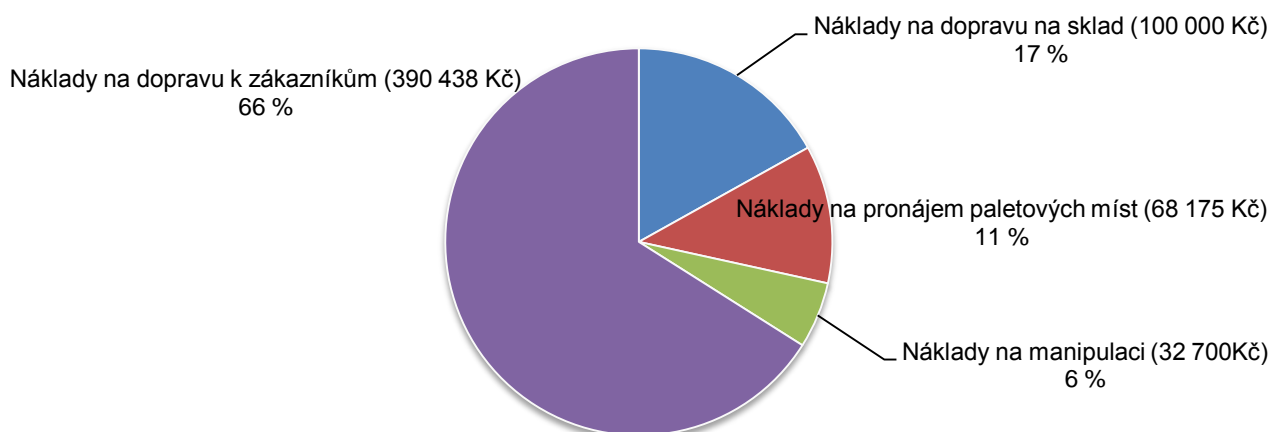
Počet dovezených palet doporučeného řešení	545 palet/rok
Počet dovezených palet na Moravu v roce 2014	334 palet/rok

Tabulka 16 obsahuje doporučené rozdělení dodávek kamionem společnosti Zach Trans, s.r.o. z Dobřejovic na sklad do Brna. V závorkách jsou uvedeny doporučené velikosti zásilek (počet palet). Cílem optimalizace pomocí stochastického modelu zásob bylo distribuování zboží po 26 paletách, neboli po plných kamionech.

Tab. 16: Doporučené rozdělení dodávek do skladu v Brně-Štýřice [autor]

		Levell Duo Plus	Baukleber	Levell Uni	QS
Období	Útlum 1	1.12. (26 p <sup>41</sup> )	1.12. (16 p)	1.12. (19 p)	1.12. (26 p)
		1.4. (26 p) 1.5. (26 p) 1.6. (26 p)	1.4. (26 p)	1.4. (26 p) 16.5. (26 p)	
	Útlum 2	1.7. (26 p) 22.7. (26 p) 11.8. (26 p)	1.7. (26 p)	1.7. (26 p)	
		1.9. (26 p) 1.10. (26 p) 1.11. (26 p)	1.9. (19 p)	1.9. (26 p)	

Předpokládané roční náklady doporučeného řešení jsou znázorněny v grafu 6. Roční náklady na dopravu na sklad vychází 100 000 Kč, roční náklady na pronájem paletových míst 68 175 Kč, roční náklady na manipulaci s paletami 32 700 Kč a náklady na dopravu k zákazníkům činí 390 438 Kč.



Graf 6: Předpokládané roční náklady doporučeného řešení [autor]

<sup>41</sup> p = jednotka paleta.

## 6. Závěr

V diplomové práci byla provedena podrobná analýza logistických procesů společnosti Sto, s.r.o., hlavního výrobce a dodavatele zboží pro vnější úpravu budov. Pomocí analýzy SWOT byla nalezena slabá místa. Za proces s největšími finančními ztrátami byla sledována distribuce zboží na Moravu a do Slezska. Navrhovaným řešením tohoto problému bylo zvoleno skladování určitého množství palet malt na pronajatých paletových místech na Moravě a ve Slezsku.

Ve třetí kapitole byly popsány disciplíny operačního výzkumu, které byly použity k nalezení optimálního řešení. Jmenovitě jde o lokační analýzu, teorii grafů, vícekriteriální rozhodování, teorii zásob a lineární programování. V teoretické části diplomové práce byl kladen důraz na vysvětlení modelů, pojmů a vzorců, které byly použity v praktické části.

Za hlavní část diplomové práce lze považovat záměnnou heuristiku popsanou v kapitole o lokační analýze. Vytvořený program v MATLABu pracuje na principu iterativního algoritmu a dokázal řešit úlohu efektivně, na rozdíl od exaktní metody. Doba výpočtu exaktní metody stoupla s počtem umísťovaných dep exponenciálně. Oproti tomu iterativní algoritmus se dobou výpočtu při lokaci dep nedostal nad jednu sekundu.

Čtvrtá kapitola se zabývá postupným výpočtem úlohy a optimalizací logistického řetězce společnosti Sto, s.r.o. Úloha byla z velké části řešena pomocí vytvořených programů, které byly napsány na základě teorie uvedené ve třetí kapitole diplomové práce.

V 1. fázi výpočtu byly určeny váhy vrcholů dopravní sítě pomocí vícekriteriálního rozhodování. Poté bylo možné použít program řešící úlohu lokační analýzy pomocí iterativního algoritmu. Úloha lokační analýzy jednoho skladu určila umístění skladu v Brně. Lokací dvou skladů bylo vypočítáno umístění skladů v Brně a Ostravě. Lokace tří skladů určila sklady ve městech Brno, Ostrava a Zlín.

Po provedení fáze lokační analýzy byly vypočteny jednotlivé okružní jízdy. Tato úloha spadající do teorie grafů určila náklady na dopravu z nových skladů k zákazníkům. Náklady na dopravu k zákazníkům jsou 390 438 Kč a tvoří 66 % z celkových nákladů doporučeného řešení. Celkové náklady na nová řešení zahrnují i náklady na skladování, které byly vypočítány pomocí stochastického modelu zásob. Do nákladů na skladování byla navíc mimo pronájmu paletových míst a poplatků za manipulaci s paletami započítány i náklady na dopravu z Dobřejovic na nový sklad.

V závěru diplomové práce bylo stanoveno doporučené řešení pro firmu Sto, s.r.o. Doporučené řešení je varianta s pronajmutím paletových míst ve skladu Brno – Štýřice s vynecháním materiálu Levell Duo. Náklady na dopravu na sklad vychází 100 000 Kč/rok. Náklady na pronájem paletových míst v Brně tvoří 11 % s hodnotou 68 175 Kč. Finanční úspora doporučeného řešení je 21,3 % z nákladů na distribuci malt firmou Sto, s.r.o. v roce 2014 na Moravu a do Slezska.

## 7. Použité zdroje

### 7.1 Literatura

- [1] EMMETT, S.: *Řízení zásob*. Computer Press, Brno, 2008, ISBN 978-80-251-1828-3.
- [2] BRÁZDOVÁ, M.: *Řešené úlohy lineárního programování*. Universita Pardubice, Pardubice, 2011, ISBN 978-80-7395-361-4.
- [3] CORDEAU, J.; LAPORTE, G.: *Vehicle Routing*, Montreal, 2007.
- [4] DEMEL, J.: *Grafy a jejich aplikace*. Praha, Academia, 2002, ISBN 80-200-0990-6.
- [5] DRAHOTSKÝ, I.; ŘEZNÍČEK, B.: *Logistika. Procesy a jejich řízení*. Brno, Computer Press, 2003, ISBN 80-7226-521-0.
- [6] JABLONSKÝ, J.: *Operační výzkum. Kvantitativní modely pro ekonomické rozhodování*. Praha, 2002, ISBN 80-86419-42-8.
- [7] JANÁČEK, J.: *Optimalizace na dopravních sítích*. Žilina, 2006, ISBN 80-8070-586-0.
- [8] JIRSÁK, P.; MERVART, M.; VINŠ, M.: *Logistika pro ekonomy – vstupní logistika*. Praha, 2012, ISBN 978-80-7357-958-6.
- [9] JUROVÁ, M.: *Podniková logistika*. Brno, 1993.
- [10] KOZEL, P.: *Úloha okružních jízd s časovými okny*. Vysoká škola báňská, Ostrava, 2011.
- [11] LINDA, B.: *Stochastické modely operačního výzkumu*. Bratislava, Statis, 2010, ISBN 978-80-85659-59-7.
- [12] LIU, W.; LIN, CH.; CHIU, CH.; TSAO, Y.; WANG, Q.: *Minimizing the Carbon Footprint for the Time-Dependent Heterogeneous-Fleet Vehicle Routing Problem with Alternative Paths*. Nová Tchaj-pej, 2014.
- [13] LUKÁŠ, L.: *Pravděpodobnostní modely v managementu*. Praha, Academia ČMT, 2012, ISBN 978-80-200-2005-5.
- [14] ŘÍHA, Z.; TICHÝ, J.; HONCŮ, M.; KUNST, J.: *Ekonomika a řízení podniku*. ČVUT v Praze, Praha, 2009, ISBN 978-80-01-04434-6.
- [15] SEDLÁČEK, J.: *Úvod do teorie grafů*. Praha, Academia, 1981.
- [16] VOLEK, J.; LINDA, B.: *Teorie grafů - aplikace v dopravě a veřejné správě*. Univerzita Pardubice, Pardubice, 2012, ISBN – 978-80-7395-225-9.

### 7.2 Internetové zdroje

- [17] BRAINTOOLS. SWOT analýza. *Braintools.cz* [online]. [vid. 2016-01-20]. Dostupné z: <http://www.braintools.cz/toolbox/strategie/swot-analyza.htm>.
- [18] EVROPSKÁ DATABANKA. Sto, s.r.o. *Edb.cz* [online]. [vid. 2016-02-10]. Dostupné z: <http://www.edb.cz/firma-326371-sto-dobrejovice>.

- [19] IPODNIKATEL. SWOT analýza. *Ipodnikatel.cz* [online]. [vid. 2016-01-20]. Dostupné z: <http://www.ipodnikatel.cz/Marketing>.
- [20] OBEC DOBŘEJOVICE. Obec v číslech. *Dobrejovice.eu* [online]. [vid. 2016-01-25]. Dostupné z: <http://www.dobrejovice.eu/obec-v-cislech>.
- [21] STO, s.r.o. Vize a mise. *Sto.cz* [online]. [vid. 2015-09-5]. Dostupné z: [http://www.sto.cz/142284\\_CZ-O\\_spolecnosti-Vize\\_a\\_mise.htm](http://www.sto.cz/142284_CZ-O_spolecnosti-Vize_a_mise.htm).
- [22] VYSOKÁ ŠKOLA BÁŇSKÁ. Floydův algoritmus. *Vsb.cz* [online]. [vid. 2016-02-10]. Dostupné z: [home1.vsb.cz/~dor028/Floydov\\_algoritmus.doc](http://home1.vsb.cz/~dor028/Floydov_algoritmus.doc).
- [23] ŠAFRÁNEK, J.: Systémové inženýrství a rozhodování (přednáška) Praha: ČVUT v Praze, 2014.

## 8. Seznam obrázků

- |           |                                                                   |
|-----------|-------------------------------------------------------------------|
| Obrázek 1 | Výstup programu Open Door Logistics Studio                        |
| Obrázek 2 | Graf                                                              |
| Obrázek 3 | Popis iterativního algoritmu                                      |
| Obrázek 4 | Popis pozměněného iterativního algoritmu                          |
| Obrázek 5 | Porovnání TSP a VRP                                               |
| Obrázek 6 | Stochastický model zásob se signalizací změn a ztracenými prodeji |
| Obrázek 7 | Výstup programu                                                   |
| Obrázek 8 | Hlavní menu programu Open Door Logistics Studio                   |

## 9. Seznam grafů

- |        |                                                               |
|--------|---------------------------------------------------------------|
| Graf 1 | Prodané množství zboží v letech 2010-2015                     |
| Graf 2 | Křivka poptávky po materiálu v roce 2014                      |
| Graf 3 | Prodané množství malt společností Sto, s.r.o. v roce 2014     |
| Graf 4 | Grafické znázornění úlohy VHV                                 |
| Graf 5 | Exponenciální pokles dopravní práce v závislosti na počtu dep |
| Graf 6 | Předpokládané roční náklady doporučeného řešení               |

## 10. Seznam tabulek

- |           |                                 |
|-----------|---------------------------------|
| Tabulka 1 | Náklady na dopravu k zákazníkům |
| Tabulka 2 | Náklady na dopravu na sklad     |
| Tabulka 3 | Tabulka cen za km               |
| Tabulka 4 | Analýza SWOT firmy Sto, s.r.o.  |

Tabulka 5	Porovnání množství prodeje
Tabulka 6	Prodané množství malt na Moravě a ve Slezsku v roce 2014 a náklady na dopravu
Tabulka 7	Seřazení měst
Tabulka 8	Porovnání metod stanovení vah kritérií
Tabulka 9	Porovnání výsledků použitých metod
Tabulka 10	Tabulka typů vozidel
Tabulka 11	Vypočtené náklady na dopravu k zákazníkům
Tabulka 12	Výsledky stochastického modelu zásob
Tabulka 13	Vypočtené náklady na dopravu k zákazníkům a na skladování
Tabulka 14	Celkové náklady nových řešení
Tabulka 15	Shrnutí doporučeného řešení
Tabulka 16	Doporučené rozdělení dodávek do skladu v Brně-Štýřice

## 11. Seznam příloh

Příloha 1	Výpočet Floydova algoritmu (zdrojový kód MATLAB)
Příloha 2	Výpočet úlohy lokační analýzy pomocí iterativního algoritmu
Příloha 3	Fullerova metoda
Příloha 4	Zdrojový kód VHV úlohy
Příloha 5	Zdrojový kód úlohy lokační analýzy
Příloha 6	Teorie zásob – vypočtené hodnoty malt
Příloha 7	Zdrojový kód stochastického modelu zásob

## 12. Přílohy

### Příloha 1 – Výpočet Floydova algoritmu (zdrojový kód MATLAB)

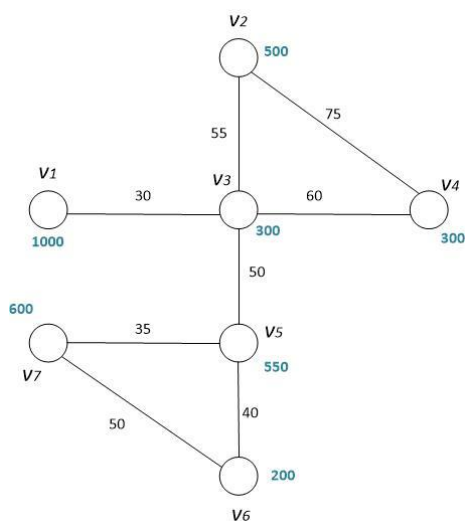
```
s=(1 2 2 2 3 3);           %Zadání vrcholů 1 (předchůdců)
t=(2 3 4 5 4 5);           %Zadání vrcholů 2 (následovníků)
w=(3 5 6 4 2 3);           %o(h)(ohodnocení hran)
G=graph(s,t,w);             %Vytvoření grafu
d=distances(G);             %Floydův algoritmus
```

Řešení z konzole MATLABu (distanční matice)

```
0  3  8  9  7
3  0  5  6  4
8  5  0  2  3
9  6  2  0  5
7  4  3  5  0
```

### Příloha 2 – Výpočet úlohy lokační analýzy pomocí iterativního algoritmu

**Zadání:** Určete vrcholově optimální rozmístění tří dep. Ohodnocení hran označuje délku komunikací mezi vrcholy. Váhy vrcholů vyjadřují množství stavebního materiálu [kg] vezeného do daného města během jednoho dne.



Výpočet úlohy budeme provádět dle obecně zapsaného algoritmu krok za krokem.

**1. krok:** Zvolíme si výchozí množinu třech dep  $D_3 = \{v_1, v_3, v_5\}$ , tato volba není náhodná. Vrchol  $v_1$  je zvolen kvůli nejvyšší váze. Naopak vrcholy  $v_3$  a  $v_5$  jsou zvoleny kvůli jejich nejvyššímu stupni ( $st(v_3) = 4, st(v_5) = 3$ ). Pomocná proměnná  $z = 0$  a množina neprozkoumaných vrcholů nyní obsahuje rozdíl množiny všech vrcholů grafu a množiny  $D_3$  tedy  $N = V \setminus D_k$ . Nyní musíme vyřešit atrakční obvody, přičemž se musíme řídit definicí přiděleného atrakčního obvodu.

$$A^*(v_1) = \{v_1\}; A^*(v_3) = \{v_2, v_3, v_4\}; A^*(v_5) = \{v_5, v_6, v_7\};$$

Nyní musíme vypočítat dopravní práci pro jednotlivé atrakční obvody.

Výpočet vychází z kritéria pro optimální rozmístění dep na síti (obsluha vrcholů).

$$\sum_{u \in A^*(v_1)} 2 * d(u, v_1) * w(u) = 0$$

$$\sum_{u \in A^*(v_3)} 2 * d(u, v_3) * w(u) = 2 * 60 * 300 + 2 * 55 * 500 = 91\ 000$$

$$\sum_{u \in A^*(v_5)} 2 * d(u, v_5) * w(u) = 2 * 40 * 200 + 2 * 35 * 600 = 58\ 000$$

Celková dopravní práce  $f(D_3) = 149\ 000$ .

**2. krok:** Množina neprozkoumaných vrcholů  $N \neq \emptyset$  a proto musíme dále postupovat dle kroku 2b, tedy vybereme vrchol ( $v_2$ ) z množiny  $N$  a vytvoříme množiny  $D_3^{v_j}$ . Ty jsou vytvářeny tak, že postupně nově vybraným vrcholem nahradíme všechny tři vrcholy z aktuálního řešení.

$$D_3^{v_1} = \{v_2, v_3, v_5\}$$

$$D_3^{v_3} = \{v_1, v_2, v_5\}$$

$$D_3^{v_5} = \{v_1, v_3, v_2\}$$

Nyní musíme každý vrchol těchto nově vzniklých množin jednoznačně zařadit do přidělených atrakčních obvodů.

$$1) D_3^{v_1} = \{v_2, v_3, v_5\}$$

$$A^*(v_2) = \{v_2\}; A^*(v_3) = \{v_1, v_3, v_4\}; A^*(v_5) = \{v_5, v_6, v_7\}.$$

$$2) D_3^{v_3} = \{v_1, v_2, v_5\}$$

$$A^*(v_1) = \{v_1, v_3\}; A^*(v_2) = \{v_2, v_4\}; A^*(v_5) = \{v_5, v_6, v_7\}.$$



$$3) D_3^{v_5} = \{v_1, v_3, v_2\}$$

$$A^*(v_1) = \{v_1\}; A^*(v_3) = \{v_3, v_4, v_5, v_6, v_7\}; A^*(v_2) = \{v_2\}.$$

Množinám odpovídají následující hodnoty dopravní práce:

$$f(D_3^{v_1}) = 2 * 30 * 1000 + 2 * 60 * 300 + 2 * 40 * 200 + 2 * 35 * 600 = 154\ 000,$$

$$f(D_3^{v_3}) = 2 * 30 * 300 + 2 * 75 * 300 + 58\ 000 = 121\ 000,$$

$$f(D_3^{v_5}) = 2 * 60 * 300 + 2 * 50 * 550 + 2 * 90 * 200 + 2 * 85 * 600 = 229\ 000.$$

$$A \text{ určíme } f(D_k^{v_r}) = f(D_3^{v_3}) = \min\{154\ 000, 121\ 000, 229\ 000\}$$

**3. krok:**  $f(D_3) > f(D_3^{v_3})$ , musíme tedy postupovat dle kroku 3b. Což znamená, že pomocná proměnná  $z = z + 1$ , tedy  $z = 1$  a vytvoříme novou množinu dep  $D_k = \{v_1, v_2, v_5\}$  a pokračujeme opět krokem číslo 2.

**2. krok:** V tento moment množina neprozkoumaných vrcholů obsahuje vrcholy  $v_4, v_6$  a  $v_7$ . Jako další vrchol ke kombinování se současnou množinou  $D_k$  vybereme například vrchol  $v_4$ .

$$D_3^{v_1} = \{v_4, v_2, v_5\}$$

$$D_3^{v_2} = \{v_1, v_4, v_5\}$$

$$D_3^{v_5} = \{v_1, v_2, v_4\}$$

Opět musíme nyní určit jednotlivé atrakční obvody.

$$1) D_3^{v_1} = \{v_4, v_2, v_5\}$$

$$A^*(v_4) = \{v_4\}; A^*(v_2) = \{v_2\}; A^*(v_5) = \{v_1, v_3, v_5, v_6, v_7\}.$$

$$2) D_3^{v_2} = \{v_1, v_4, v_5\}$$

$$A^*(v_1) = \{v_1, v_3\}; A^*(v_4) = \{v_2, v_4\}; A^*(v_5) = \{v_5, v_6, v_7\}.$$

$$3) D_3^{v_5} = \{v_1, v_2, v_4\}$$

$$A^*(v_1) = \{v_1, v_3\}; A^*(v_2) = \{v_2, v_4\}; A^*(v_4) = \{v_5, v_6, v_7\}.$$

Množinám odpovídají následující hodnoty dopravní práce:

$$f(D_3^{v_1}) = 2 * 50 * 300 + 2 * 80 * 1000 + 58\ 000 = 248\ 000,$$

$$f(D_3^{v_2}) = 2 * 30 * 300 + 2 * 75 * 500 + 58\ 000 = 151\ 000,$$

$$f(D_3^{v_5}) = 2 * 30 * 300 + 2 * 80 * 550 + 2 * 120 * 200 + 2 * 115 * 600 = 292\ 000.$$

A určíme  $f(D_k^{v_r}) = f(D_3^{v_2}) = \min\{248\ 000, 151\ 000, 292\ 000\}$ .

**3. krok:** Dále postupujeme dle kroku 3a, protože  $f(D_3) < f(D_3^{v_2})$ . Vrchol  $v_4$  nepoužijeme, odstraníme ho z množiny  $N$  a pokračujeme krokem 2.

**2. krok:** Nyní z množiny neprozkoumaných vrcholů použijeme další vrchol, třeba  $v_6$ .

$$D_3^{v_1} = \{v_6, v_2, v_5\}$$

$$D_3^{v_2} = \{v_1, v_6, v_5\}$$

$$D_3^{v_5} = \{v_1, v_2, v_6\}$$

Opět musíme nyní určit jednotlivé atrakční obvody

$$1) D_3^{v_1} = \{v_6, v_2, v_5\}$$

$$A^*(v_6) = \{v_6\}; A^*(v_2) = \{v_2, v_4\}; A^*(v_5) = \{v_1, v_3, v_5, v_7\}.$$

$$2) D_3^{v_2} = \{v_1, v_6, v_5\}$$

$$A^*(v_1) = \{v_1, v_2, v_3, v_4\}; A^*(v_6) = \{v_6\}; A^*(v_5) = \{v_5, v_7\};$$

$$3) D_3^{v_5} = \{v_1, v_2, v_6\}$$

$$A^*(v_1) = \{v_1, v_3\}; A^*(v_2) = \{v_2, v_4\}; A^*(v_6) = \{v_5, v_6, v_7\}.$$

Množinám odpovídají následující hodnoty dopravní práce:

$$f(D_3^{v_1}) = 2 * 75 * 300 + 2 * 50 * 300 + 2 * 35 * 600 + 2 * 80 * 1000 = 232\ 000,$$

$$f(D_3^{v_2}) = 2 * 30 * 300 + 2 * 90 * 300 + 2 * 85 * 500 + 2 * 35 * 600 = 199\ 000,$$

$$f(D_3^{v_5}) = 2 * 30 * 300 + 2 * 75 * 300 + 2 * 40 * 550 + 2 * 35 * 600 = 235\ 000.$$

A určíme  $f(D_k^{v_r}) = f(D_3^{v_2}) = \min\{232\ 000, 199\ 000, 235\ 000\}$ .

**3. krok:** Dále postupujeme dle kroku 3a, protože  $f(D_3) < f(D_3^{v_2})$ . Vrchol  $v_6$  nepoužijeme, odstraníme ho z množiny  $N$  a pokračujeme krokem 2.

**2. krok:** Nyní z množiny neprozkoumaných vrcholů použijeme poslední vrchol, vrchol  $v_7$ .

$$D_3^{v_1} = \{v_7, v_2, v_5\}$$

$$D_3^{v_2} = \{v_1, v_7, v_5\}$$

$$D_3^{v_5} = \{v_1, v_2, v_7\}$$

Opět musíme nyní určit jednotlivé atrakční obvody.

$$1) D_3^{v_1} = \{v_7, v_2, v_5\}$$

$$A^*(v_7) = \{v_7\}; A^*(v_2) = \{v_2, v_4\}; A^*(v_5) = \{v_1, v_3, v_5, v_6\}.$$

$$2) D_3^{v_2} = \{v_1, v_7, v_5\}$$

$$A^*(v_1) = \{v_1, v_2, v_3, v_4\}; A^*(v_7) = \{v_7\}; A^*(v_5) = \{v_5, v_6\}.$$

$$3) D_3^{v_5} = \{v_1, v_2, v_7\}$$

$$A^*(v_1) = \{v_1, v_3\}; A^*(v_2) = \{v_2, v_4\}; A^*(v_7) = \{v_5, v_6, v_7\}.$$

Množinám odpovídají následující hodnoty dopravní práce:

$$f(D_3^{v_1}) = 2 * 75 * 300 + 2 * 50 * 300 + 2 * 80 * 1000 + 2 * 40 * 200 = 251\ 000,$$

$$f(D_3^{v_2}) = 2 * 30 * 300 + 2 * 90 * 300 + 2 * 85 * 500 + 2 * 40 * 200 = 173\ 000,$$

$$f(D_3^{v_5}) = 18\ 000 + 45\ 000 + 58\ 500 = 121\ 500.$$

$$A \text{ určíme } f(D_k^{v_r}) = f(D_3^{v_5}) = \min\{251\ 000, 173\ 000, 121\ 500\}.$$

**3. krok:** Dále postupujeme dle kroku 3a, protože  $f(D_3) < f(D_3^{v_2})$ . Vrchol  $v_6$  nepoužijeme, odstraníme ho z množiny  $N$  a pokračujeme krokem 2.

**2. krok:** Řídíme se krokem 2a, protože množina neprozkoumaných vrcholů je prázdná ( $N = \emptyset$ ) a pokračujeme tedy krokem 4.

**4. krok:** Dle 4b ( $z > 0$ ) musíme nyní položit  $z = 0$  a určíme novou množinu neprozkoumaných vrcholů  $N = V \setminus D_k$ . Aktuální řešení úlohy, teda umístění dep ve vrcholech  $v_1, v_2$  a  $v_5$  považujeme za výchozí množinu dep  $D_k$ . Dále pokračujeme krokem 2 a celý postup algoritmu opakujeme. Jestliže se už řešení v průběhu výpočtu nezmění a nezmění se tedy ani pomocná proměnná  $z$ , přejdeme na krok 5, který ukončuje algoritmus. Dostaneme tak (sub)optimální rozmístění  $k$  dep na síti.

## 2. Řešení úlohy lokační analýzy pomocí MATLABu

Číslo/a vrcholu/ů, kam sklady umístít:

1 3 5                      %počáteční množina

Dopravní práce pro tuto variantu:

149000

Číslo/a vrcholu/ů, kam sklady umístít:

1 2 5                      %přechod algoritmu k lepšímu řešení

Dopravní práce pro tuto variantu:

121000

Atrakční obvody

1                      % $A^*(v_1) = \{v_1, v_3\}$

1 0 3 0 0 0 0

2                      % $A^*(v_2) = \{v_2, v_4\}$

0 2 0 4 0 0 0

5                      % $A^*(v_5) = \{v_5, v_6, v_7\}$

0 0 0 0 5 6 7

### Příloha 3 – Fullerova metoda

Fullerova metoda	4	K1	K2	K3	K4	Suma	Váha
Množství materiálu	K1	0	1	1	1	3	0,50
Počet objednávek	K2	0	0	1	1	2	0,33
Cena	K3	0	0	0	0,5	0,5	0,08
výdálčnost od Tollwitu	K4	0	0	0,5	0	0,5	0,08
					<b>Součet</b>	5,5	1,00

### Příloha 4 – Zdrojový kód VHV úlohy

```
A = xlsread('C:\Users\vojta\Desktop\Iterativnialgoritmus\vahy.xls',1);
%Načtení excelu.
c=length(A(1,:));
w=length(A(:,1));
for i=1:c
    k(i,1)=input('Zadejte hodnotu pořadové funkce kritéria k(i,kde
i=1,...,počet kritérií):');
    k2(i,1)=input('Je kritérium v poměrové(1) nebo pořadové stupnici(0)?');
    if k2(i,1)==0
        k1(i,j)=1;
        continue;
    end;
    k1(i,1)=input('Je kritérium maximalizační(1) nebo minimalizační(0)?');
```

```

end;

%Stanovení vah kritérií
for i=1:c
    b(i,1)=(2*(k(i,1))/(c*(c+1)));
end;
D=sum(b);
kontrola=input('Provést kontrolu vah kritérií?');

switch kontrola
    case 1
        disp('Součet vah kritérií je roven:');
        disp(D)
    case 0
end;

%Převod kritérií z minimalizačních na maximalizační
for i=1:w
    for j=1:c
        D=max(A(:,j));
        D1=min(A(:,j));
        if k1(j,1)==1;
            Q(i,j)=A(i,j);
        else Q(i,j)=(D*D1)/A(i,j);
        end;
    end;
end;

%Normované váhy kritérií
for j=1:c
    for i=1:w
        D=max(Q(:,j));
        Q1(i,j)=(Q(i,j))/D;
    end;
end;

%Vážené normované hodnoty
for j=1:c
    for i=1:w
        Q2(i,j)=Q1(i,j)*b(j,1);
    end;
end;

%Součet řádků
for i=1:w
    Q3(i,1)=sum(Q2(i,:));
end;
disp(Q3)
xlswrite('C:\Users\vojta\Desktop\Iterativnialgoritmus\vahy2.xls',Q3)

```

## **Příloha 5 – Zdrojový kód úlohy lokační analýzy**

```

clc;
clearvars;
disp('----- (C)Vojtěch Pazdera, FD ČVUT, Praha-----')
disp('-----')
disp('-----Lokační analýza, Iterativní algoritmus-----')
disp('-----')
disp('Dobrý den, Vítejte v programu Lokační analýzy.')

```

```

tic;
iterativhrub=input('Přejete si k Vašemu výpočtu použít Algoritmus Hrubé
síly (1), nebo Iterativní algoritmus (2)?');
v=input('Máte vloženou distanční matici?');

[num] =
xlsread('C:\Users\vojta\Desktop\Iterativnialgoritmus\vahy2.xls',1,'A:A');
%Načtení vyh vrcholů z excelu.
weights1=num;
weights1=weights1';
y1=length(weights1(1,:));

switch v
case 1 %Vytváření grafu z excelu.
% d
=importdata('C:\Users\vojta\Desktop\Iterativnialgoritmus\DistancniMaticet.txt');
%Distanční matice z textového souboru.
[num] =
xlsread('C:\Users\vojta\Desktop\Iterativnialgoritmus\dm.xls',1,'A:A');
s=num;
s=s';
[num] =
xlsread('C:\Users\vojta\Desktop\Iterativnialgoritmus\dm.xls',1,'B:B');
t=num;
t=t';
[num] =
xlsread('C:\Users\vojta\Desktop\Iterativnialgoritmus\dm.xls',1,'C:C');
weights=num;
weights=weights';
G=graph(s,t,weights); %Vykreslení grafu.
%
plot(G,'EdgeLabel',G.Edges.Weight,'NodeColor','k','EdgeColor','k')
d=distances(G); %Floydův algoritmus pomocí funkce distnaces.

case 0 %Případ pro zadávání grafu ručně.
x=input('Kolik hran má Váš graf?');

for i = 1:x
s(1,i)=input('Zadej první vrchol:');
t(1,i)=input('Zadej druhý vrchol:');
weights(1,i)=input('Zadej jejich vzdálenost');
end;
G=graph(s,t,weights); %Vytvoření grafu.
%
plot(G,'EdgeLabel',G.Edges.Weight,'NodeColor','k','EdgeColor','k')
d=distances(G);
end;
% disp(d);

k=input('Kolik dep chcete umístit?');

%Množina kandidátů
z11=input('Přejete si zadat množinu kandidátů?');
if z11==1
z2=input('Kolik vrcholů množina kandidátů obsahuje?');
for i=1:z2
z(1,i)=input('Zadejte číslo vrcholu kandidáta:');
end;

```

```

elseif z11==0
    y=length(d(1,:));
    z=1:y;
end;

%Deklarace proměnných
n=length(d(1,:));n1=n;
pole=1:k; %Počáteční kombinace.
v=0; D11=Inf;
y=0;a=zeros(1,n);m=length(z(:,1));b=zeros(1,n);pole1=zeros(1,k);
D111=0;

if k>n %Uživatel zadal lokaci více dep, než má graf vrcholů.
    disp('Zadáno chybně, program bude okamžitě ukončen!!')
    quit;
end;

%uložení výsledků z Command Window
vystup=input('Chcete uložit data z průběhu výpočtu z Command Window?');
switch vystup
    case 1
        diary on
    case 0
end;
disp(d)

%Lokace jednoho depa se vždy provádí hrubou silou.
if k==1
    iterativhrub=1;
end;

%Algoritmus hrubé síly
switch iterativhrub
    case 1

tic;
while 1>0
y2=1;
mA=ismember(pole,z); %Speciální funkce, bere jen ty prvky, které jsou
v obou vektorech.
if sum(mA)~=k %Pokud není mA stejně dlouhý vektor jako k, breakneme
cyklus while.
    break;
end;
for i=1:k %Řešení pro první možnou kombinaci k vrcholů
    D=d(pole(1,i),:);
    b(i,:)=D;
end;
for j=1:n
    v=min(b(:,j));
    if v==0
        continue;
    end;
    D=(2*(v*(weights1(j)))); %Výpočet dopravní práce
    a(1,j)=a(1,j)+D;
end;

D111=sum(a);
disp('Číslo/a vrcholu/ů, kam sklady umístít:')
disp(pole) %Výpis té varianty, jejíž dopravní práce je min

```

```

        pole1=pole;
        disp('Dopravní práce pro tuto variantu:')
        disp(D111)           %Dopravní práce této varianty
break;
end;
    D11=D111;
    a=zeros(1,n);        %Vynulování a

while 1>0
y2=y2+1;
    for i=k:-1:1        %Generování možných kombinací vrcholů
        if pole(i)<n-(k-i)
            pole(i)=pole(i)+1;
            for j=i+1:k
                pole(j)=pole(j-1)+1;%Po průchodu všech kombinací break.
            end;
            break;
        end;
    end;

mA=ismember(pole,z);
if sum(mA)~=k
    continue;
end;

if z11==1;
    n1=length(z(1,:));
end;

for i=1:k
    D=d(pole(1,i),:);
    b(i,:)=D;
end;

    for j=1:n
        v=min(b(:,j));
        if v==0
            continue;
        end;
        D=(2*(v*(weights1(j))));    %Výpočet dopravní práce
        a(1,j)=a(1,j)+D;
    end;

    D1=sum(a);
    if D11>=D1
        D11=D1;
        disp('Číslo/a vrcholu/ů, kam sklady umístit:')
        disp(pole)           %Výpis té varianty, jejíž
dopravní práce je min.
        disp('Dopravní práce pro tuto variantu:')
        disp(D1)           %Dopravní práce této varianty
        pole1=pole;
        end;
        a=zeros(1,n);

        if pole(1)==n1-k+1    % Opět po průchodu všech kombinací
vrcholů break.
            break;
        end;
end;
end;

```



```

%Iterativní algoritmus
case 2
disp('Nyní zadejte vrcholy počáteční množiny. ');
for i=1:k
    a1(i)=input('Zadejte číslo vrcholu: ');
end;

tic;
while 1>0

pole=a1;
b1=1:y1;

%deklarace
vysledek=0;k=length(a1(1,:));g=zeros(1,n);
D3=0;cyklus=0;D2=0;y2=1; z1=0; n=length(b1(1,:));n1=n;

if z1==1 %Vybíráme pouze prvky z kandidátů.
    b1=z;
    n1=length(b1(1,:));
end;

c1=ismember(b1,a1); %Vytváříme vektor 1 a 0 (Abychom z vektoru
brali pouze neprozkoumané vrcholy).
for i=1:n1
    if c1(1,i)==0
        vysledek(1,end+1)=b1(1,i); %A tyto prvky si zapisujeme do pomocné
proměnné vysledek.
    elseif c1(1,i)==1
        continue;
    end;
end;
vysledek1=vysledek(2:end); %Ořiznutí 0.
pole1=a1;
for i=1:k %Jako u alg. hrubé síly i zde počítám počáteční množinu zvlášť.
    D=d(pole(1,i),:); %V další části totiž rovnou беру prvky
neprozkoumané množiny b1.
    b(i,:)=D;
end;
for j=1:n
    v=min(b(:,j));
    if v==0
        continue;
    end;
    D=(2*(v*(weights1(j)))); %Výpočet dopravní práce.
    g(1,j)=g(1,j)+D;
end;

D111=sum(g);
disp('Číslo/a vrcholu/ů, kam sklady umístít: ')
disp(pole) %Výpis té varianty, jejíž dopravní práce je min.
disp('Dopravní práce pro tuto variantu: ')
disp(D111) %Dopravní práce této varianty.
g=zeros(1,n);

D11=D111;
for x=1:(n1-k)
    y2=y2+1;
    for j=1:k
        a=a1;
        D3=D11;
    end;
end;

```

```

        pole(1,j)=vysledek1(1,x);
        for i=1:k
            D=d(pole(1,i),:);
            b(i,:)=D;
        end;

    for y=1:n
        v=min(b(:,y));
        if v==0
            continue;
        end;
        D=(2*(v*(weights1(y))));
        g(1,y)=g(1,y)+D;
    end;

        D1=sum(g);
        D2(1,end+1)=D1;
        pole=a;
        g=zeros(1,n);
    end;

    D22=D2(2:end);
    [r,c] = find(D22==min(D22(:)));      %Hledání minima z vektoru D22,
    nejmenší dopravní práce.
    D3=r;
    D3=min(D22);
        if D11>=D3
            z1=z1+1;
            D11=D3;
            pole(1,c)=vysledek1(1,x);
            a=pole;
            pole1=a;
            disp(a)
            disp(D3)
        end;
        pole=a;
        D3=0;
        D2=0;
    end;

    if z1>0
        a1=pole1;
    elseif z1==0
        break;
    end;
    end;

end;
switch k
    case 1
        u=1:n;
            disp(pole1)
            disp(u(1,:))
        otherwise
    end

%Atrakční obvody pro lokalizované sklady
n=length(d(1,:));
u=zeros(k,n);      %Vynulování matic u a b, příprava místa
b=zeros(k,n);

```

```

for i=1:k
    D=d(pole1(1,i),:);
    b(i,:)=D;
end;
for j=1:n
    v=min(b(:,j));
    for i=1:k
        if v~=b(i,j)           %Vytvoření matice b (obsahuje pouze 1 a 0)
            b(i,j)=0;
        elseif v==b(i,j)
            b(i,j)=1;
        end;
    end;
end;
for i=1:k
    for j=1:n
        if b(i,j)==1
            u(i,j)=u(i,j)+j;
        end;
    end;
end;
%Vytvoření matice u (Atrakční obvody)
disp('Atrakční obvody')
for i=1:k
    disp(pole1(1,i))
    disp(u(i,:))
end;
end;

% disp('Počet iterací')
% disp(y2)

%výstup
h=plot(G,'EdgeLabel',G.Edges.Weight);
set(gca,'XDir','reverse'); %Osy grafu.
% set(gca,'YDir','reverse');
set(gca,'visible','off');

    highlight(h,(1:n))           %Nastavení velikosti vrcholů.
%    highlight(h,[pole1])
    highlight(h,[pole1])
for i=1:k
    u1=0;
    for j=1:n
        if u(i,j)==0
            continue;
        elseif u(i,j)>0
            u1(1,end+1)=u(i,j);
        end;
    end;
    u11=u1(2:end);
    highlight(h,[u11],'NodeColor',rand(1,3)) %Atrakční obvody.
end;
toc;
%uložení výsledků z Command Window
switch vystup
case 1
    diary('Vystup.txt')
    edit('Vystup.txt')
    disp('Výsledky uloženy. Konec čtení programu.')

```

```

case 0
disp('Konec čtení programu.')
end;

```

### Příloha 6 – Teorie zásob – vypočtené hodnoty malt

<b>Útlum 1</b>	<b>Levell duo plus</b>			<b>Levell duo</b>			<b>Baukleber</b>			<b>Levell Uni</b>			<b>QS</b>		
<b>k</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>
m	25	36	45				16	24	30	19	28	36	26	38	48
h	3	6	9				3	4	6	3	6	3	8	14	18
m'	14	20	27				9	14	18	11	16	21	17	24	30
n	1	2	3				1	2	3	1	2	3	1	2	3
<b>Špička 1</b>	<b>Levell duo plus</b>			<b>Levell duo</b>			<b>Baukleber</b>			<b>Levell Uni</b>			<b>QS</b>		
<b>k</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>
m	26	52	78	26	52	78	25	36	45	26	52	66			
h	8	14	18	24	44	63	3	6	9	5	8	12			
m'	19	34	48	34	30	84	14	20	24	17	30	39			
n	3	2	3	5	6	6	1	2	3	2	2	3			
<b>Útlum 2</b>	<b>Levell duo plus</b>			<b>Levell duo</b>			<b>Baukleber</b>			<b>Levell Uni</b>			<b>QS</b>		
<b>k</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>
m	26	52	78	26	52	78	25	36	45	26	38	48			
h	18	44	60	23	40	60	3	6	9	3	4	6			
m'	29	58	78	31	54	81	14	18	21	14	22	27			
n	3	4	3	2	2	3	1	2	3	1	2	3			
<b>Špička 2</b>	<b>Levell duo plus</b>			<b>Levell duo</b>			<b>Baukleber</b>			<b>Levell Uni</b>			<b>QS</b>		
<b>k</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>1</b>	<b>2</b>	<b>3</b>
m	26	52	78	26	52	78	19	28	33	26	28	36			
h	9	14	18	22	38	57	2	4	6	3	6	6			
m'	20	36	51	29	50	75	11	16	18	15	16	21			
n	3	4	6	2	4	6	1	2	3	1	2	3			

### Příloha 7 - Zdrojový kód stochastického modelu zásob

```

clear all %#ok<CLALL>
b=(5/3)*4; %průměrnáspotřeba

q1=6500; %cena za dopravu
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
q2=3.75*120; %cena zaskladování
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

q4=550;

a1=1; %Spodní hranice rovnoměrného rozdělení psti
a2=5; %Horní hranice rovnoměrného rozdělení psti

```

```

rozdil=a2-a1;

m(1,1)=sqrt((2*b*q1)/q2);
h(1,1)=a2-rozdil*((q2*m(1,1))/(q4*b+q2*m(1,1)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
EU=((h(1,1)^2)/8)-(5/4)*h(1,1)+25/8;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

m(1,2)=sqrt((2*b*(q1+q4*EU)/q2));

disp(m(1,1))
disp(m(1,2))
disp(h(1,1))

for x=2:10
h(1,x)=a2-rozdil*((q2*m(1,x))/(q4*b+q2*m(1,x)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
EU=((h(1,x)^2)/8)-(5/4)*h(1,x)+25/8;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

m(1,x+1)=sqrt((2*b*(q1+q4*EU)/q2));
disp('-----');
disp(m(1,x))
disp(m(1,x+1))
disp(h(1,x))
disp('-----');
end;
m=m(1,10);
h=h(1,10);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
ES=(25/8)-(h^2)/8;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

m_prumer=ceil((m+h-ES+EU+h-ES+EU)/2);
n=ceil(b/m);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% EC=(m/b)*60;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

disp('Průměrné množství palet na skladě')
disp(m_prumer)
disp('Střední počet dodávkových cyklů')
disp(n)
% disp('Střední doba cyklu')
% disp(EC)% disp('Cena za skladování')
% % disp(c)

```