

UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA

BAKALÁŘSKÁ PRÁCE

Softwarové nástroje pro podporu vícekriteriálního
hodnocení a rozhodování



Katedra matematické analýzy a aplikací matematiky

Vedoucí bakalářské práce: **doc. RNDr. Jana Talašová, CSc.**

Vypracoval(a): **Dagmar Kubíková**

Studijní program: B1103 Aplikovaná matematika

Studijní obor Matematika–ekonomie se zaměřením na bankovníctví/pojišťovnictví

Forma studia: prezenční

Rok odevzdání: 2016

BIBLIOGRAFICKÁ IDENTIFIKACE

Autor: Dagmar Kubíková

Název práce: Softwarové nástroje pro podporu vícekriteriálního hodnocení a rozhodování

Typ práce: Bakalářská práce

Pracoviště: Katedra matematické analýzy a aplikací matematiky

Vedoucí práce: doc. RNDr. Jana Talašová, CSc.

Rok obhajoby práce: 2016

Abstrakt: Cílem bakalářské práce je popsat aktuální stav v oblasti softwarových nástrojů pro podporu vícekriteriálního rozhodování. Práce se zabývá základní charakteristikou vybraných softwarových nástrojů, vysvětlením nejužívanějších a nejznámějších metod hodnocení a ukázkou práce ve vybraných programech na praktickém příkladu. V bakalářské práci jsou zmíněny čtyři rozhodovací metody – AHP, ELECTRE III, PROMETHEE a metoda vícekriteriální funkce utility. V závěru je ve třech programech (D-Sight, 1000Mind, v Make It Rational) řešen praktický příklad, který se týká výběru rodinného automobilu.

Klíčová slova: vícekriteriální rozhodování, softwarové nástroje, AHP, ELECTRE III, PROMETHEE, metoda aditivní vícekriteriální funkce utility

Počet stran: 62

Počet příloh: 0

Jazyk: český

BIBLIOGRAPHICAL IDENTIFICATION

Author: Dagmar Kubíková

Title: Software tools for support of multiple criteria evaluation and decision making

Type of thesis: Bachelor's

Department: Department of Mathematical Analysis and Application of Mathematics

Supervisor: doc. RNDr. Jana Talašová, CSc.

The year of presentation: 2016

Abstract: The main goal of this bachelor thesis is to describe the nowadays software tools for multiple criteria decision making. The thesis starts with basic characteristics of chosen software tools. Then the author aims at explanation of the most used and the most popular methods of multiple-criteria decision making; four decision-making methods (AHP, ELECTRE III, PROMETHEE and the method of the additive multiple criteria utility function) are described in detail in this bachelor thesis. At the end the solution of a practical example (buying a family car) by means of three different software tools (D-Sight, 1000Mind, Make It Rational) is presented.

Key words: multiple criteria decision making, software tools, AHP, ELECTRE III, PROMETHEE, the method of additive multiple criteria utility function

Number of pages: 62

Number of appendices: 0

Language: Czech

Prohlášení

Prohlašuji, že jsem bakalářskou práci zpracovala samostatně pod vedením paní doc. RNDr. Jany Talašové, CSc. a všechny použité zdroje jsem uvedla v seznamu literatury.

V Olomouci dne

.....

podpis

Obsah

Úvod	8
1 Vícekriteriální rozhodování	9
2 Softwarové nástroje	11
2.1 1000Minds	11
2.2 ChemDecide	12
2.3 Decisionarium	13
2.3.1 „Windows software“	13
2.3.2 „Web software“	13
2.4 DEXi	14
2.5 D-Sight	14
2.6 ElectioVis	15
2.7 M-MACBETH	16
2.8 Make It Rational	16
2.9 Expert Choice	16
2.10 PriEsT	17
3 Popis vybraných metod	18
3.1 AHP – Analytický hierarchický proces	18
3.1.1 Tvorba hierarchie	19
3.1.2 Párové srovnávání	19
3.1.3 Saatyho bodovací stupnice	20
3.1.4 Saatyho matice	20
3.1.5 Vlastní čísla a vlastní vektory reciprokových matic	22
3.1.6 Geometrický průměr řádků Saatyho matice	23
3.1.7 Konzistence	23
3.1.8 Hodnocení variant	25
3.1.9 Vyhodnocení nejlepší varianty	26
3.2 ELECTRE III	26
3.2.1 Matice intenzit párových preferencí V	27
3.2.2 Charakteristika D	27
3.2.3 Sestupné pořadí variant	28

3.2.4	Vzestupné preferenční uspořádání variant	29
3.3	PROMETHEE	29
3.3.1	Povaha kritérií	30
3.3.2	Diference	30
3.3.3	Míra preference	30
3.3.4	Parametry	31
3.3.5	Preferenční funkce	31
3.3.6	Agregované preference	34
3.3.7	Toky ohodnocení	35
3.3.8	PROMETHEE I	36
3.3.9	PROMETHEE II	36
3.4	Metoda vícekritériální funkce utility	37
3.4.1	Konstrukce vícekritériální funkce utility	37
3.4.2	Ověření preferenční nezávislosti kritérií	38
3.4.3	Konstrukce dílčích funkcí utility	39
3.4.4	Stanovení vah dílčích utilit kompenzační metodou	40
3.4.5	Celková funkce utility v aditivním tvaru	42
4	Představení vybraných systémů	43
4.1	D-Sight	44
4.1.1	Praktický příklad řešený v D-Sight	45
4.2	1000Minds	48
4.2.1	Praktický příklad řešený v 1000Minds	49
4.3	Make It Rational	53
4.3.1	Praktický příklad řešený v Make It Rational	53
	Závěr	57
	Seznam obrázků	59
	Seznam tabulek	60
	Literatura	61

Poděkování

Ráda bych poděkovala vedoucí bakalářské práce doc. RNDr. Janě Talašové, CSc. za odborné vedení, spolupráci, čas a cenné připomínky, které mi věnovala při konzultacích. Ráda bych také poděkovala mé rodině za všestrannou podporu, kterou mi během studia poskytla.

Úvod

Tématem bakalářské práce jsou softwarové nástroje pro podporu vícekritériálního hodnocení a rozhodování. Rozhodování je důležitou součástí každodenního života fyzických i právnických osob. Mnohé firmy usilují o optimální rozhodnutí v nejrůznějších situacích, a proto si zakoupí či vytvoří software, jenž rozhodování usnadní. Neexistují však pouze programy komerční, ale i softwary určené ke studijním účelům.

Hlavním cílem bakalářské práce je popsat aktuální stav v oblasti softwarových nástrojů pro podporu vícekritériálního rozhodování. V současné době existuje velké množství takovýchto programů, a proto bylo nutno vybrat pouze několik z nich. V práci jsou uvedeny systémy nejužívanější či pro svou metodu nejzajímavější. V krátkosti budou také zmíněny programy, které jsou zajímavé z historického hlediska. Dalším cílem je popsat důležité a známé metody, na kterých jsou některé softwary založeny.

Práce je rozdělena do čtyř kapitol. V první části jsou uvedeny základní pojmy z oblasti vícekritériálního rozhodování. Další kapitola se věnuje jednotlivým softwarovým nástrojům. Nejde zde prioritně o detailní analýzu programů, ale spíše o základní charakteristiku systémů. Lze-li takové informace dohledat, je u softwaru uveden důvod vzniku, tvůrci a rozhodovací metoda, kterou systém užívá. Třetí kapitola se zabývá popisem vybraných metod. Poslední část práce je zaměřena prakticky. Byly vybrány tři softwary a pomocí nich byl vyhodnocen konkrétní rozhodovací problém.

Kapitola 1

Vícekriteriální rozhodování

Tato kapitole je zpracována podle zdroje [1].

V reálném životě se člověk často ocitne v situaci, která nabízí více než jedno přípustné řešení. Takovou situací může být například výběr vhodného automobilu, restaurace pro narozeninovou oslavu, dodavatele či vhodného zaměstnance. Aby mohl být problém vyřešen, musí mít rozhodovatel k dispozici množinu kritérií, která mu umožní posuzovat jednotlivá řešení, a množinu alternativ, ze které vybírá tu nejlepší variantu.

Proces rozhodování lze zapsat takto: „Je dána množina n variant $X = \{x_1, x_2, \dots, x_n\}$, které jsou posuzovány dle m stanovených hledisek, kritérií z množiny $K = \{K_1, K_2, \dots, K_m\}$. Úkolem je vybrat z dané množiny variant X variantu x^* , která je nejlepší vzhledem ke kritériím z množiny K “ [1, s. 85].

V teorii rozhodování se vyskytuje množství specifických pojmů, z nichž ty nejdůležitější budou nyní v krátkosti vysvětleny. Pojmem rozhodnutí se rozumí výběr z několika alternativ, které rozhodovatel zvažuje. Rozhodovací situace je situace, kdy je třeba se rozhodnout. Rozhodovatel je účastník rozhodovací situace, který se rozhoduje. Rozhodovací proces je proces řešení rozhodovacích problémů. Optimálním řešením (variantou, alternativou) je nazváno rozhodnutí pro rozhodovatele nejlepší.

Velké množství metod je založeno na tom, že celkové hodnocení se počítá jako vážený průměr dílčích hodnocení. Společný charakter těchto metod je dán

následujícím schématem:

$$K = \{K_1, \dots, K_m\}, X = \{x_1, \dots, x_n\}, v_j \geq 0, j = 1, \dots, m, \sum_{j=1}^m v_j = 1$$

$$\forall x \in X : h(x) = \sum_{j=1}^m v_j h_j(x).$$

K ... množina nezávislých kritérií, předpoklad nezávislosti viz [1],

X ... množina alternativ,

v_j ... normovaná váha j -tého dílčího hodnocení, váha kritéria K_j ,

$h(x)$... celkové hodnocení varianty x ,

$h_j(x)$... dílčí hodnocení varianty x podle kritéria $K_j, j = \{1, \dots, m\}$.

Metody se liší v tom, jak jsou váhy a dílčí hodnocení interpretovány a ve způsobu výpočtu dílčích hodnocení. Mezi metody, které jsou na tomto principu založeny patří například metoda bazické varianty, metoda statistické standardizace, bodovací metoda, metoda univerzální standardizace, metoda dílčích cílů. Teoreticky propracovaným přístupem váženého průměru dílčích hodnocení je metoda vícekritériální funkce utility a také Saatyho Analytický hierarchický proces. K určení optimální varianty využívají váhy kritérií i další metody, například ELECTRE III nebo PROMETHEE.

Metody pracují s konkrétními naměřenými hodnotami kritérií (reálná čísla), ale jsou i takové, které jsou schopny pracovat s neurčitější informací, tzn. není zadáno jedno číslo, ale například interval nebo tři hodnoty (předpokládaná nejvyšší, předpokládaná nejnižší, nejvíce očekávaná).

Kapitola 2

Softwarové nástroje

Ruční výpočet nejlepší varianty z množiny všech alternativ může být někdy komplikovaný a dnes se v praxi nepoužívá. Jednou z variant, jak se vyhnout ručnímu výpočtu nejlepší alternativy je použít MC Excel. V dnešní době jsou však k dispozici propracované softwary, které podporují vícekriteriální rozhodování, usnadní rozhodovateli práci při konstrukci optimálního řešení a poskytnou uživateli věcné a jasné podklady pro rozhodnutí. Některé z nich jsou představeny v této kapitole. Programů pro podporu vícekriteriálního rozhodování existuje velká řada, zde jsou popsány ty nejužívanější či z pohledu použité metody nejzajímavější. Zmíněny jsou i programy, které vznikly jako jedny z prvních a to proto, aby bylo vidět jaké využívaly metody. U každého softwaru je uvedeno, na jaké metodě se zakládá. Seznamy dostupných programů pro řešení úloh vícekriteriálního rozhodování lze nalézt na webové stránce <http://www.mcdmsociety.org>. Tento seznam je po několika letech vždy aktualizován. Další seznam je například na stránce: <https://en.wikipedia.org/wiki/>.

2.1. 1000Minds

Kapitola je zpracována podle zdroje [2].

Software 1000Minds byl vytvořen Paulem Hansen a Franzem Oblerem v pozdních devadesátých letech na Univerzitě v Otagu na Novém Zélandu. Software užívá k vyhodnocení optimální varianty metodu PAPRIKA. Blíže je software

popsán v kapitole 4.2.

2.2. ChemDecide

Tato kapitola je zpracována podle zdroje [3].

Software ChemDecide byl vyvinut Richardem Hodgettem na univerzitě Newcastle ve Velké Británii v rámci projektu Britest.

Název projektu Britest je zkratkou pro Best Route Innovative Technology Evaluation and Selection Techniques. Tento projekt byl započat na základě společného výzkumu, jenž byl realizován předními chemicky a farmaceuticky orientovanými společnostmi z Velké Británie ve spolupráci s akademickými institucemi. Cílem projektu bylo vytvořit řadu inovativních nástrojů a metod pro analýzu a vývoj produktů, stejně jako výrobních procesů. Roku 2001 se projekt rozšířil i za hranice Velké Británie.

Software ChemDecide byl poprvé komerčně použit v roce 2014 při výběru zařízení, financování a skladování chemikálií. Díky velké flexibilitě lze software užít téměř pro všechny typy rozhodovacích problémů.

ChemDecide není volně dostupný pro komerční účely, využívat jej mohou pouze členové projektu Britest. ChemDecide obsahuje čtyři softwarové nástroje, jeden pro strukturování rozhodovacího problému (Decision Setup) a další tři pro jeho vyhodnocení (ChemDecide-AHP, ChemDecide-MARE and ChemDecide-RANK). ChemDecide umí pracovat jak s kvantitativními, tak s kvalitativními informacemi.

Richard Hodgett vytvořil video, které ukazuje, jak Chemdecide pracuje. Prezentovaným problémem je výběr kávovaru. Video je umístěno na webové stránce <https://www.youtube.com/watch?v=HzBnEhfC4F4>.

ChemDecide umožňuje použít jednu ze tří metod vícekriteriálního rozhodování:

- ChemDecide–AHP – využívá metodu AHP, což je zkratkou pro Analytický hierarchický proces,

- ChemDecide-MARE – používá metodu MARE, což je zkratka pro Multi-Attribute Range Evaluations,
- ChemDecide-RANK – využívá metodu ELECTRE III.

2.3. Decisionarium

Kapitola a její podkapitoly jsou vypracovány ze zdroje [4].

Decisionarium je dostupné na webové stránce Aalto University ve Finsku. „Windows software“ a „web software“ byly vyvinuty studenty a vývojáři této školy.

2.3.1. „Windows software“

Decisionarium nabízí dva „windows software“, které lze na webových stránkách Aalto University zdarma stáhnout. Jedná se o softwary WINPRE a PRIME Decisions, které využívají stejnojmenné metody. Oba softwary byly vyvinuty na začátku devadesátých let, a nepatří tedy k nejnovějším. Tyto programy nemusí tudíž na dnešních moderních počítačích vždy správně fungovat. Programy neppracují s přesnými hodnotami kritérií, ale s intervalovými. V dnešní době je k dispozici modernější software a pro rozhodovací problém vhodnější metody, proto zde WINPRE a PRIME Decisions nejsou podrobněji rozebrány.

2.3.2. „Web software“

Decisionarium nabízí především webové aplikace, které jsou modernější než-li „windows software“. Jedná se např. o software: Web-HIPRE a Smart-Swaps.

Web-HIPRE

Web-HIPRE je webová verze softwaru HIPRE 3+. Tyto dva programy byly jedny z prvních internetových aplikací podporujících vícekritériální rozhodování. HIPRE 3+ pracuje na základě dvou metod: AHP a SMART. Lze pracovat s jednotlivými metodami odděleně nebo je kombinovat. Webová verze nabízí více me-

tod, než software HIPRE 3+, kterými lze rozhodovací problém vyřešit: SWING, SMART, SMARTER a AHP.

Smart–Swaps

Webová aplikace Smart–Swaps je založena na metodě Even Swaps. Smart–Swaps je k dispozici online pro akademické využití.

2.4. DEXi

Kapitola je vypracována podle zdroje [5].

Software DEXi byl vyvinut ve spolupráci Jožef Stefan Institute a Faculty of Organizational Sciences ve Slovinsku. Vývoj DEXi byl zahájen v roce 1999 a vznikl za dvojnásobným účelem. Prvním důvodem byla obecná snaha koncipovat modernější nástupce softwaru DEX. Dalším motivem byla snaha rozšířit program ve školství. V roce 1999 se program DEX začal vyučovat v několika univerzitních kurzech a objevila se idea zavést metodiku DEX také na střední školy. Proto vznikla silná potřeba počítačového programu jednoduchého na obsluhu. V následujících letech se DEXi rozšířil nejen do školství, ale byl použit i pro mnoho projektů. Tento software je vhodný nejen pro vzdělávání, ale také na podporu řešení reálných rozhodovacích problémů. Program DEXi je založen na stejnojmenné metodě.

První verze DEXi 1.00 byla vydána v roce 2000 s jediným slovinským uživatelským prostředím. V roce 2016 vznikla poslední verze DEXi 5.02, kterou je možno zdarma stáhnout v anglickém jazyce či ve slovinštině.

2.5. D-Sight

Kapitola je vypracována podle zdroje [6].

Software D-Sight vznikl na základě podrobného zkoumání metody PROMETHEE. Quantin Hayez, výzkumník z Computer & Decision Engineering Department na Univerzitě Libre v Bruselu, vyvinul software D-Sight založený právě

na této metodě a s uplatněním přístupu vícekriteriální funkce utility. Software D-Sight je podrobněji popsán v kapitole 4.1.

2.6. ElectioVis

Kapitola popsána ze zdroje [7].

ElectioVis je software, který uživatele ihned po navštívení oficiálních webových stránek vyzve, aby se do softwaru zaregistroval. Bez tohoto kroku je další pohyb na webových stránkách nemožný. Bohužel zde nejsou uvedeny žádné obecné informace o ElectioVis. Webové stránky slouží přímo k tvorbě rozhodovacího modelu.

V prvé řadě uživatel pojmenuje svůj rozhodovací problém a podrobněji jej definuje. V dalším kroku uživatel zodpoví na sedm uzavřených otázek, díky kterým software vybere vhodnou metodu k řešení problému. Otázky lze vidět na obrázcích 2.1 a 2.2. Software využívá tři metody, kterými vyhodnotí nejlepší variantu: AHP, Electre III a metodu vícekriteriální funkce utility.

1) What would you like to obtain at the end of the process?

- Anumerical score for each of the options I am choosing among.
- Aranking of the options will be enough (e.g. option "C" is the best, option "A" is the second best, etc.).
- I only want to assign each option to one or more categories (e.g. "Recommendable options", "Options to be rejected", etc.).

2) If an option is very good in one aspect but very bad in another aspect, would you like to make compensations?

- Not at all.
- Yes, they should be compensated.

3) In which way would you like to define your preferences?

- I would like to do comparisons such as "X is strongly better in Quality than Y", "they have the same Comfort", etc.
- I would like to define all my preferences directly with numbers.
- I would like to use both categories ("Good", "Bad", etc.) and numbers.

4) How stable is the list of options from where you are choosing?

- Once I define them, I am sure they will not change.
- They might change but, if they do, I will create a new decision.
- They might change and, if they do, I want to keep consistency without creating a new decision.

Obrázek 2.1: Uzavřené otázky

5) Will you need to explain how you arrived to the recommended choice to people who are not familiar with decision aiding?

Yes.

No.

6) While comparing alternatives, you will use different criteria. Those criteria...

... are totally independent from each other.

... might have some interdependencies with each other.

7) I would like to use a model that..

... relies on strict rationality rules.

... advises the best choice according to my system of values.

Obrázek 2.2: Uzavřené otázky

2.7. M-MACBETH

Software popsán ze zdroje [8].

Software M-MACBETH je založen na metodě MACBETH. Software vytváří vizualizaci rozhodovacího problému ve formě stromové struktury.

2.8. Make It Rational

Software popsán ze zdroje [9].

Make It Rational je software, který byl vyvinut v Polsku v Poznani. Software využívá pro hodnocení nejlepší varianty metodu AHP. Make It Rational je podrobněji popsán v kapitole 4.3.

2.9. Expert Choice

Software popsán ze zdroje [10].

Software Expert Choice byl vyvinut v roce 1983 Ernestem Formanem na George Washington University ve Washingtonu DC. Software využívá pro určení optimální alternativy metodu AHP. Mezi společnostmi, které využívají software Expert Choice patří například NASA.

2.10. PriEsT

Software popsán ze zdroje [\[11\]](#).

PriEsT, což je zkratkou pro Priority Estimation Tool, je volně stažitelný software. Tento program vyvinul Sajid Siraj ve spolupráci s University of Portsmouth. Software využívá pro určení nejlepší varianty metodu AHP.

Kapitola 3

Popis vybraných metod

V této kapitole jsou popsány vybrané metody, na kterých jsou některé z představených programů založeny. Popsané metody jsou jedny z nejdůležitějších a nej-používanějších v oblasti matematické podpory vícekritériálního rozhodování.

3.1. AHP – Analytický hierarchický proces

Kapitola a její podkapitoly jsou vypracovány ze zdrojů [12], [13].

V roce 1980 byla tato metoda navržena T.L. Saatyem, podle něj je také často označována jako Saatyho metoda. Základem metody je rozklad složité nestrukturované situace na jednodušší komponenty – vytvoření hierarchického systému. Díky této hierarchii má rozhodovatel větší přehled o řešeném problému. Výhodou AHP je to, že dokáže pracovat jak s kvalitativními důsledky variant vzhledem ke kritériím, tak s kvantitativními.

Metoda AHP se řadí mezi metody párového porovnávání. Párové srovnávání nespočívá u této metody pouze v určování preferencí mezi kritérii, ale také v určování intenzit těchto preferencí. Rozhodovatel má k dispozici bodovací stupnici se slovními popisy, podle které určuje, jak moc je jedno kritérium významnější než druhé. Poté porovnává jak moc je jedna varianta lepší než druhá vzhledem ke každému kritériu. Jeho rozhodování o preferencích by mělo být vždy konzistentní. Pokud je vypočtená konzistence dostatečná, může být pomocí získaných vah kritérií a vypočtených hodnocení variant vytvořeno celkové hodnocení vari-

ant.

3.1.1. Tvorba hierarchie

Členění umožňuje rozhodovateli celkový pohled na problém. Problém se rozdělí na jednotlivá hlediska, která mohou ovlivňovat dosažení celkového cíle. Hlediska mohou být dále také členěna a tím je tvořena složitější hierarchie. Po rozdělení problému na jednotlivá části, které má smysl spolu porovnávat, získá rozhodovatel přehlednou strukturu, díky které je schopen říci, zda do rozhodování zahrnul všechno, co by mohlo ovlivňovat dosažení cíle.

Nejtypičtější a nejjednodušší hierarchie obsahuje tři úrovně.

- 1. úroveň – cíl neboli rozhodovací problém,
- 2. úroveň – kritéria, podle kterých jsou posuzovány jednotlivé varianty,
- 3. úroveň – varianty, ze kterých vybíráme tu nejlepší.

Pokud lze jednotlivá kritéria na druhé úrovni více rozčlenit a vytvořit tak tzv. subkritéria, vznikne v hierarchii další úroveň. Z každého kritéria může vycházet libovolný počet subkritérií. Tímto způsobem lze vytvořit libovolný hierarchický systém. Zde bude uvažována pouze základní třístupňová hierarchie.

3.1.2. Párové srovnávání

Po vytvoření hierarchické struktury rozhodovacího problému je možno přistoupit k určení vah kritérií a zjištění dílčích hodnocení variant vzhledem k jednotlivým kritériím, čehož dosáhneme pomocí párového srovnávání.

Hlavnímu cíli, který je na první hierarchické úrovni se přiřadí váha 1. Tuto váhu je třeba rozdělit mezi všechna kritéria $K_1, K_2, K_3, \dots, K_m$ na druhé hierarchické úrovni. Pomocí párového srovnávání jednotlivých kritérií vzhledem k celkovému cíli jsou určeny váhy $w_1, w_2, w_3, \dots, w_m$ pro tato kritéria, ty jsou následně znormovány na váhy $v_1, v_2, v_3, \dots, v_m$. Nyní jsou porovnány varianty $x_1, x_2, x_3, \dots, x_n$, na třetí hierarchické úrovni, vzhledem ke každému kritériu.

Poté je pro varianty určeno hodnocení $h_j(x_i)$, pro každé $j = 1, 2, 3, \dots, m$, $i = 1, 2, 3, \dots, n$. Kde $h_j(x_i)$ znamená hodnocení i -té varianty vzhledem k j -tému kritériu.

Pokud by ke každému kritériu $K_1, K_2, K_3, \dots, K_m$ byly váhy $w_1, w_2, w_3, \dots, w_m$ známé, může být sestavena matice W relativních preferencí kritérií, která má následující tvar:

$$W = \begin{pmatrix} \frac{w_1}{w_1} & \frac{w_1}{w_2} & \dots & \frac{w_1}{w_m} \\ \frac{w_2}{w_1} & \frac{w_2}{w_2} & \dots & \frac{w_2}{w_m} \\ \frac{w_3}{w_1} & \frac{w_3}{w_2} & \dots & \frac{w_3}{w_m} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{w_m}{w_1} & \frac{w_m}{w_2} & \dots & \frac{w_m}{w_m} \end{pmatrix}$$

Váhy $w_1, w_2, w_3, \dots, w_m$ nejsou ovšem při rozhodování známé, proto je zadávána matice relativních preferencí kritérií, kterou můžeme chápat jako odhad matice W , a váhy jsou z ní dopočítány.

3.1.3. Saatyho bodovací stupnice

Aby bylo možné porovnávat jednotlivá kritéria mezi sebou z hlediska toho, jak moc je jedno kritérium významnější než druhé, je nutné určit stupnici, na které se bude toto porovnávání provádět. T. L. Saaty zavedl devítibodovou stupnici, kterou doplnil slovním popisem.

V tabulce 3.1 jsou uvedeny jazykově popsané hodnoty ze základní devítibodové Saatyho stupnice. Základní hodnoty, které se nejčastěji využívají pro párové hodnocení jsou 1, 3, 5, 7, 9. Pro jemnější určení významnosti jednoho kritéria nad druhým, lze použít i hodnoty 2, 4, 6, 8, ty však nemají českou slovní interpretaci.

3.1.4. Saatyho matice

Kritéria $K_1, K_2, K_3, \dots, K_m$ se párově porovnají s použitím prvků Saatyho škály a výsledky se zapíší do tzv. Saatyho matice S , která má tvar:

hodnotící stupeň	porovnání prvků x a y	vysvětlení
1	x je stejně významný jako y	oba prvky přispívají stejnou měrou k cíli
3	x je mírně významnější než y	zkušenosti a úsudek mírně preferují první prvek před druhým
5	x je silně významnější než y	existuje silná preference prvního prvku před druhým
7	x je velmi silně významnější než y	existuje velmi silná preference prvního prvku před druhým
9	x je absolutně významnější než y	skutečnosti upřednostňující první prvek před druhým mají nejvyšší stupeň průkaznosti

Tabulka 3.1: Jazykově popsané prvky Saatyho škály

$$S = \begin{pmatrix} s_{11} & s_{12} & \dots & s_{1m} \\ s_{21} & s_{22} & \dots & s_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ s_{m1} & s_{m2} & \dots & s_{mm} \end{pmatrix}.$$

Prvek s_{ij} , který nabývá hodnot ze základní devítibodové Saatyho škály, vyjadřuje pro každé $i, j = 1, 2, \dots, m$, že kritérium K_i je s_{ij} -krát významnější než kritérium K_j , tedy významnost kritéria K_j představuje $1/s_{ij}$ -tou část významnosti kritéria K_i . U Saatyho matice tedy předpokládáme, že je reciproká, tj. pro všechna $i, j, i, j = 1, 2, 3, \dots, m$, musí platit:

$$s_{ij} = \frac{1}{s_{ji}},$$

potom je matice S reciproká.

Pokud jsou porovnávána kritéria K_i a $K_j, i, j = 1, 2, \dots, m$, na pozici prvku s_{ij} se doplní hodnota z devítibodové Saatyho škály $\{1, 2, 3, \dots, m\}$. Kritérium K_j nemusí být porovnáváno vzhledem ke kritériu K_i , na pozici prvku s_{ji} se vepíše převrácená hodnota prvku s_{ij} .

Z praktického hlediska je vhodné kritéria nejprve uspořádat od nejvýznamnějšího po nejméně významné. Tento krok není nutný, ale usnadní zadavateli dodržení konzistence matice S . Tento krok pouze usnadní rozhodovateli přehlednost

v matici S . V horním trojúhelníku matice S se budou nacházet pouze hodnoty $\{1, 2, 3, \dots, 9\}$.

Matice S , při jejímž zadávání byl rozhodovatel dostatečně konzistentní ve svých výpovědích, viz kapitola 3.1.7, je odhadem matice W a platí:

$$s_{ij} \approx \frac{w_i}{w_j}.$$

Ukázka Saatyho matice:

	K_1	K_2	K_3	\dots	K_m
K_1	1	3	5	\dots	9
K_2	1/3	1	3	\dots	5
K_3	1/5	1/3	1	\dots	5
\dots	\dots	\dots	\dots	1	\dots
K_m	1/9	1/5	1/5	\dots	1

Tabulka 3.2: Příklad tvorby Saatyho matice

3.1.5. Vlastní čísla a vlastní vektory reciprokových matic

K určení vah kritérií je nutné spočítat vlastní vektor odpovídající maximálnímu vlastnímu číslu λ_{max} matice párových srovnávání S . Řešením soustavy m rovnic o m neznámých $\mathbf{w} = (w_1, w_2, w_3, \dots, w_m)$ ve vektorovém tvaru:

$$(S - \lambda_{max}I)\mathbf{w} = \mathbf{0},$$

jinak vyjádřeno:

$$S\mathbf{w} = \lambda_{max}\mathbf{w},$$

kde $\mathbf{0}$ je sloupcový nulový vektor o m prvcích, λ_{max} je maximální vlastní číslo matice S , I je jednotková matice, se získá vlastní vektor \mathbf{w} . Vlastní čísla a jim odpovídající vlastní vektory se získají v Matlabu pomocí příkazu $eig(S)$.

Pokud má reciproká matice všechny prvky kladné, potom má maximální kladné reálné vlastní číslo λ_{max} . Vlastní vektor odpovídající vlastnímu číslu λ_{max} má všechny prvky se stejným znaménkem. Pokud se jedná o znaménko minus, lze vlastní vektor vynásobit číslem (-1) . Z tohoto plyne, že pro Saatyho matici lze

považovat vlastní vektor odpovídající λ_{max} za vektor nenormovaných vah. Platí to jen v případě, že matice S je dosti konzistentní, viz kapitola 3.1.7.

Normované váhy kritérií, tedy takové, které dají v součtu číslo jedna, se získají za pomoci vlastního vektoru \mathbf{w} :

$$v_j = \frac{w_j}{\sum_{j=1}^m w_j}, j = 1, 2, \dots, m.$$

3.1.6. Geometrický průměr řádků Saatyho matice

Matice S je odhadem matice W , proto je možné při výpočtu vah postupovat také tak, že se bude minimalizovat rozdíl mezi jejich prvky s_{ij} a $\frac{w_i}{w_j}$, pro $i, j = 1, 2, \dots, m$. Váhy w_1, w_2, \dots, w_m se pak mohou hledat minimalizací součtu čtverců:

$$\sum_{i=1}^m \sum_{j=1}^m \log(s_{ij} - \frac{w_i}{w_j})^2 \rightarrow \min$$

za podmínky:

$$w_j \geq 0, j = 1, 2, \dots, m.$$

Řešením úlohy jsou geometrické průměry řádků matice S , tj. pro $j = 1, 2, \dots, m$ platí:

$$w_j = \sqrt[m]{\prod_{k=1}^m s_{jk}}.$$

Poté, co jsou vypočteny nenormované váhy w_j , musí se opět znormovat na váhy v_j , pro $j = 1, 2, \dots, m$: $v_j = \frac{w_j}{\sum_{j=1}^m w_j}$.

Tato metoda je vhodná pro rozhodovatele, který nemá k dispozici software, jenž by mu spočetl λ_{max} a vlastní vektor.

3.1.7. Konzistence

Přirozeným předpokladem pro matici intenzit preferencí je, aby platilo, že je-li kritérium K_i s_{ij} -krát významnější než kritérium K_j a kritérium K_j je s_{jk} -krát významnější než kritérium K_k , kde $i, j, k \in \{1, 2, 3, \dots, m\}$, pak by mělo platit,

že kritérium K_i je s_{ik} -krát významnější než K_k , že tedy platí $s_{ik} = s_{ij} \cdot s_{jk}$. Toto vyjadřuje následující definice konzistence.

Definice 3.1.1 ([13])

Nechť $S = \{s_{ij}\}_{i,j=1}^m$ je čtvercová matice typu $m \times m$, pro jejíž prvky platí

$$s_{ik} = s_{ij} \cdot s_{jk},$$

pro $i, j, k = 1, 2, 3, \dots, m$.

Potom řekneme, že matice S je konzistentní.

Problematika dodržení konzistence u Saatyho matice prakticky spočívá v tom, že je-li např. $s_{ij} = 3$ a $s_{jk} = 5$, pak s_{ik} by mělo být 15 a taková hodnota v Saatyho škále není. Protože ideální konzistence matice S nedosáhne, rozhodovatel bude usilovat o to, aby matice S byla alespoň „dostatečně konzistentní“.

K ověření zda bylo při zadávání intenzit párových preferencí dosaženo dostatečné konzistence, tedy toho, zda si rozhodovatel při určování preferencí mezi prvky příliš neodporoval ve svých tvrzeních, navrhl Saaty nejdříve použít index konzistence CI (z anglického consistency index), který je definován předpisem:

$$CI = \frac{(\lambda_{max} - m)}{(m - 1)},$$

kde m je počet kritérií a λ_{max} je největší vlastní číslo matice S . Pro absolutně konzistentní matici (tedy pro matici W) platí $CI = 0$, pro ostatní reciproké kladné matice je toto číslo větší než nula. Požaduje se tedy, aby index konzistence CI matice S byl co nejbližší nule. Během dalších výzkumů se ukázalo, že index CI pořád ještě závisí na řádu matice, tedy na m . K určení hranice nekonzistence, tedy okamžiku, kdy má rozhodovatel svá hodnocení přehodnotit, proto slouží tzv. poměr konzistence CR (z anglického consistency ratio) definovaný předpisem:

$$CR = \frac{CI}{RI(m)},$$

kde $RI(m)$ je tzv. náhodný index (z anglického random index), což je průměr experimentálně zjištěných CI pro náhodně generované reciproké matice řádu m ,

jejichž prvky jsou z množiny $\{\frac{1}{9}, \frac{1}{7}, \frac{1}{5}, \dots, 1, 2, 3, \dots, 9\}$. Hodnoty $RI(m)$ lze vyhodnotit pro $m = 3, 4, 5, \dots, 16$, jsou uvedeny v tabulce 3.3.

m	3	4	5	6	7	8	9
$RI(m)$	0.525	0.882	1.115	1.252	1.341	1.404	1.452
m	10	11	12	13	14	15	16
$RI(m)$	1.484	1.513	1.535	1.555	1.57	1.583	1.595

Tabulka 3.3: Indexy $RI(m)$

Pokud $CR \leq 0.1$, potom se matice S považuje za dostatečně konzistentní. Pokud matice S není dostatečně konzistentní, je potřeba ji vhodným způsobem upravit tak, aby byla.

3.1.8. Hodnocení variant

U hodnocení alternativ dle jednotlivých kritérií se postupuje analogicky jako u určování vah kritérií. Varianty x_1, x_2, \dots, x_n jsou párově porovnány vzhledem ke každému kritériu. Pro každé kritérium je vytvořena matice párových srovnání variant vzhledem ke každému kritériu, tzn. jsou vytvořeny Saatyho matice rozměru $n \times n$. Nutné je opět kontrolovat konzistenci matic.

Hodnocení alternativ $g_j(x_i)$, pro $i = 1, 2, 3, \dots, n$ vzhledem k danému kritériu K_j , kde $j = 1, 2, 3, \dots, m$, získá rozhodovatel buď metodou vlastního vektoru nebo geometrickým průměrem řádků.

Normované hodnocení $h_j(x_i)$, pro $i = 1, 2, 3, \dots, n$ alternativ se získá stejně jako u určování vah kritérií:

$$h_j(x_i) = \frac{g_j(x_i)}{\sum_{i=1}^n g_j(x_i)},$$

pro $j = 1, 2, 3, \dots, m$.

Pokud má rozhodovatel mezi kritérii nějaká kvantitativní, výpočet se pro něj zjednodušuje. V takovéto situaci se využívá hodnot, které alternativy u jednotlivých kritérií nabývají. Takto vytvořená matice párových srovnání je zcela konzistentní.

Prvky matice se vytvoří podle následujícího předpisu:

$s_{ij}^k = \frac{y_i^k}{y_j^k}$... pro kritérium K_k s rostoucí preferencí, kde y_i^k je hodnota, které nabývá i -tá varianty vzhledem ke kritériu K_k a y_j^k je hodnota, které nabývá j -tá varianty vzhledem ke kritériu K_k .

$s_{ij}^k = \frac{y_j^k}{y_i^k}$... pro kritérium K_k s klesající preferencí, kde y_j^k je hodnota, které nabývá j -tá varianty vzhledem ke kritériu K_k a y_i^k je hodnota, které nabývá i -tá varianty vzhledem ke kritériu K_k .

Pokud varianta nabývá hodnoty nula vzhledem k některému kvantitativnímu kritériu, vytvoří se vzhledem k danému kritériu matice S pomocí devítibodové Saatyho stupnice stejně jako u kvalitativních kritérií.

Poznámka 3.1.1 *Pro kvantitativní kritéria není potřeba matice sestavovat. Pro kritéria s rostoucí preferencí se znormují hodnoty, kterých alternativy vzhledem ke kritériím nabývají. Tímto postupem se získá normované dílčí hodnocení variant dle jednotlivých kritérií: $h_k(x_i) = \frac{y_i^k}{\sum_{i=1}^n y_i^k}$, kde $h_k(x_i)$ je dílčí hodnocení varianty x_i podle kritéria K_k , y_i^k je hodnota, které nabývá i -tá varianta vzhledem ke kritériu K_k . Pro kritéria s klesající preferencí se znormují převrácené hodnoty kritérií.*

3.1.9. Vyhodnocení nejlepší varianty

Pomocí vzorce:

$$h(x_i) = \sum_{j=1}^m v_j h_j(x_i),$$

pro $i = 1, 2, 3, \dots, n$, se určí celková hodnocení všech alternativ. Varianta, která nabývá největšího hodnocení $h(x_i)$ je považována za optimální. Pro přehlednost lze jednotlivé výsledky zapsat do tabulky 3.4.

3.2. ELECTRE III

Kapitola a její podkapitoly jsou vypracovány ze zdrojů [14], [15] a [16].

	K_1	K_2	K_3	\dots	K_m	$h(x_i)$
x_1	$h_1(x_1)$	$h_2(x_1)$	$h_3(x_1)$	\dots	$h_m(x_1)$	$\sum_{j=1}^m h_j(x_1)$
x_2	$h_1(x_2)$	$h_2(x_2)$	$h_3(x_2)$	\dots	$h_m(x_2)$	$\sum_{j=1}^m h_j(x_2)$
x_3	$h_1(x_3)$	$h_2(x_3)$	$h_3(x_3)$	\dots	$h_m(x_3)$	$\sum_{j=1}^m h_j(x_3)$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
x_n	$h_1(x_n)$	$h_2(x_n)$	$h_3(x_n)$	\dots	$h_m(x_n)$	$\sum_{j=1}^m h_j(x_n)$

Tabulka 3.4: Výsledná tabulka

Metodu ELECTRE III je možné realizovat ve dvou verzích, které vedou buď k sestupnému preferenčnímu uspořádání alternativ (tj. od nejlepší k nejhorší) nebo k vzestupnému preferenčnímu uspořádání alternativ (tj. od nejhorší k nejlepší). V této metodě se rozkládá množina variant rozhodování tak, že se od ní postupně oddělují indiferenční třídy alternativ. ELECTRE III slouží k agregaci hodnocení podle jednotlivých kritérií, pokud mají dílčí hodnocení ordinální charakter.

3.2.1. Matice intenzit párových preferencí V

Pro každé kritérium K_k , $k = 1, 2, 3, \dots, m$, se vytvoří matice $P_k = \{p_{ij}\}_{i,j=1}^n$:

$$p_{ij}^k = \begin{cases} 1 \\ 0 \end{cases}$$

1 se na pozici prvku p_{ij}^k zapíše, pokud je varianta x_i preferována před variantou x_j vzhledem ke kritériu K_k , 0 ve všech jiných případech.

Každá matice P_k , $k = 1, 2, 3, \dots, m$, se vynásobí vahou v_k , $k = 1, 2, 3, \dots, m$, kritéria K_k , $k = 1, 2, 3, \dots, m$. Jednotlivé váhy kritérií se získají například Saatyho metodou. Tyto matice se sečtou a výsledná matice se označí jako V . Prvky matice V představují sílu preference varianty x_i před variantou x_j , proto je také označována jako matice intenzit párových preferencí.

3.2.2. Charakteristika D

V matici intenzit párových preferencí V se najde maximální prvek v^0 :

$$v^0 = \max \{v_{jk} : x_j, x_k \in X\}$$

a prvek v^1 , který je druhý největší v matici V :

$$v^1 = \max \{v_{jk} : x_j, x_k \in X, v_{jk} < v^0\}.$$

Prvek v^1 značí práh citlivosti preference.

Označení:

- P_i^1 – počet variant, před kterými je varianta x_i preferována s prahem citlivosti v^1 ,
- Q_i^1 – počet variant, které jsou preferovány před variantou x_i s prahem citlivosti v^1 .

Hodnota charakteristiky D_i^1 se určí jako počet variant, před nimiž je varianta x_i preferována s prahem preference v^1 , od něhož se odečte počet variant, které jsou preferovány před variantou x_i s prahem preference v^1 . Tento postup lze zapsat takto:

$$D_i^1 = P_i^1 - Q_i^1,$$

kde $i = 1, 2, 3, \dots, n$.

Jinými slovy, hodnota charakteristiky D_i^1 se pro každou variantu x_i vypočte tak, že se spočítá kolikrát se v i -tém řádku vyskytla hodnota v^0 a od výsledku se odečte kolikrát se hodnota v^0 vyskytla v i -tém sloupci.

3.2.3. Sestupné pořadí variant

Varianta, která nabývá největší hodnoty charakteristiky D^1 , se oddělí od množiny variant X jako první varianta v sestupném preferenčním uspořádání variant (jednoduchá indifferenční třída). Celý postup se opakuje s podmnožinou variant Y , která vznikla zmenšením množiny X o oddělenou variantu, tzn. z matice V se odstraní příslušný řádek a sloupec.

Pokud charakteristika D^1 nabývá maximální hodnoty u více než jedné varianty, tedy u podmnožiny Z množiny X , vytvoří se z těchto variant nová matice V' odpovídající podmnožině Z . Pro každou variantu z množiny Z se stejným

způsobem spočítá charakteristika D a určí se pořadí variant. Je-li práh citlivosti roven nule, varianty mají v sestupném uspořádání stejně pořadí (víceprvková indifferenční třída). Po určení pořadí variant z podmnožiny Z se z matice V oddělí řádky a sloupce příslušné již uspořádaným variantám a celý proces se opakuje s podmnožinou Y , která vznikla zmenšením množiny X o oddělené varianty.

Celý proces se opakuje dokud nejsou uspořádány všechny varianty z množiny X .

3.2.4. Vzestupné preferenční uspořádání variant

Pokud rozhodovatel chce varianty uspořádat od nejméně vhodné po optimální, probíhá výpočet obdobným způsobem pouze s tím rozdílem, že se vždy hledá alternativa s minimální hodnotou charakteristiky D .

3.3. PROMETHEE

Tato metoda je popsána podle zdroje [17], [18], [19] a [20].

Metoda PROMETHEE je zkratkou Preference Ranking Organization Method for Enrichment Evaluations. Pomocí této metody byly řešeny např. úlohy na poli bankovníctví, investic, zdravotnictví, turismu, chemie a dalších. Metody PROMETHEE I a II mají společný základ, liší se až v poslední kroku.

Základem metod PROMETHEE I a II je párové porovnání každé dvojice alternativ podle každého kritéria. Porovnání je prováděno pomocí jedné z šesti navržených preferenčních funkcí. Funkce jsou vytvořeny tak, aby se daly použít na co nejrozmanitější reálná kritéria a pokryly tak potřeby rozhodovatele. Význam funkcí je poměrně jasně srozumitelný a po získání určité zkušenosti není tedy problematické správně nastavit požadované parametry. Podoba preferenčních funkcí je uvedena v kapitole 3.3.5.

Kritériím je nutné přiřadit váhy, kterými rozhodovatel určí míru důležitosti jednotlivých kritérií. Váhy je možné určit jakoukoli metodou, např. Saatyho metodou. Vhodné je uvažovat normované váhy v_j , $j = 1, 2, 3, \dots, m$.

3.3.1. Povaha kritérií

V metodě PROMETHEE je nutné zajistit, aby všechna kritéria, se kterými se pracuje, byla maximalizačního typu, tzn. varianta s nejvyšším ohodnocením je nejlepší a varianta s nejnižším ohodnocením je nejhorší. Převod kritéria minimalizačního typu na kritérium maximalizačního typu se dá provést tak, že hodnoty ve sloupci minimalizačního kritéria se vynásobí číslem (-1) . Převod tedy probíhá podle vzorce:

$$y'_{ij} = -y_{ij},$$

kde y_{ij} je hodnota které nabývá i -tá varianta u j -tého kritéria, y'_{ij} je přepočtená hodnota. Tento převod hodnot vede ke ztížené interpretaci hodnot, ale matematicky je zcela správný.

3.3.2. Diference

Vypočítají se rozdíly všech variant vzhledem ke všem kritériím a zjistí se, jak se od sebe jednotlivé varianty liší:

$$d_j(x_i, x_z) = y_j(x_i) - y_j(x_z),$$

pro $i, z = 1, 2, 3, \dots, n$, $j = 1, 2, 3, \dots, m$, kde $y_j(x_i)$ je hodnota, které nabývá varianta x_i u kritéria K_j a $y_j(x_z)$ je hodnota, které nabývá varianta x_z u kritéria K_j . Tyto hodnoty jsou stále v původních jednotkách jednotlivých kritérií.

3.3.3. Míra preference

Míra preference určuje ohodnocení rozdílů vypočtených výše. Pomocí vybrané preferenční funkce F_j se získají preferenční indexy $P_j(x_i, x_z)$:

$$P_j(x_i, x_z) = F_j(d_j(x_i, x_z)), \quad i, z = 1, 2, 3, \dots, n, \quad j = 1, 2, 3, \dots, m.$$

Kde F_j značí zvolenou preferenční funkci odpovídající j -tému kritériu, která je funkcí odchylky d_j mezi variantou x_i a x_z . Preferenční indexy nabývají hodnot 0 až 1, tedy:

$$0 \leq P_j(x_i, x_z) \leq 1.$$

Pokud $P_j(x_i, x_z) = 0$, znamená to, že varianta x_i je vzhledem k j -tému kritériu horší než varianta x_z . Pokud je hodnota $P_j(x_i, x_z)$ kladná, pak toto číslo znázorňuje stupeň preference varianty x_i před variantou x_z .

Převod diferencí na míry preference způsobuje, že získané hodnocení není závislé na původních jednotkách kritérií a hodnoty jsou dobře srovnatelné napříč kritérii.

3.3.4. Parametry

V závislosti na zvolené preferenční funkci je nutné zadat potřebné parametry:

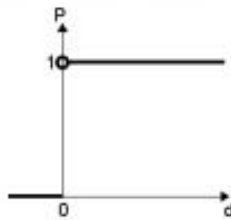
- p – práh striktní preference. Pokud je překročen, preference je 1.
- q – práh indiference. Pokud není překročen, preference je 0. Určuje zanedbatelnou odchylku variant.
- σ – parametr pro šestou preferenční funkci. Udává polohu inflexního bodu Gaussovy funkce.

3.3.5. Preferenční funkce

Průběh a zápis preferenčních funkcí:

- Preferenční funkce č. 1: přímé srovnání

U této funkce není zapotřebí žádného volitelného parametru. Přímé srovnání se nejvíce hodí pro případy, kdy jakýkoliv rozdíl mezi variantou x_i a x_z znamená absolutní preference.

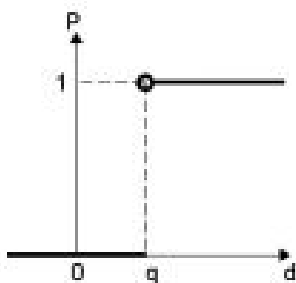


Obrázek 3.1: Průběh preferenční funkce č. 1

$$P(d) = \begin{cases} 0, & d \leq 0 \\ 1, & d > 0 \end{cases}$$

- Preferenční funkce č. 2: tvar U

Tato funkce je podobná funkci přímého srovnání. Na rozdíl od přímého srovnání zde však existuje tzv. indifferenční oblast. Tedy oblast, kde hodnota odchylky je zanedbatelná a tedy neznamená preferenci. Šíře indifferenční oblasti je dána parametrem q .



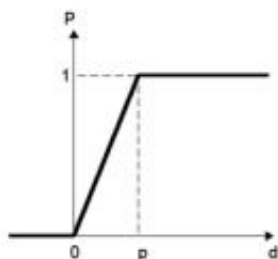
Obrázek 3.2: Průběh preferenční funkce č. 2

$$P(d) = \begin{cases} 0, & d \leq q \\ 1, & d > q \end{cases}$$

- Preferenční funkce č. 3: lineární

Obrázek 3.3 zobrazuje průběh funkce kritéria s lineární preferencí. Podle předpisu je patrné, že pokud jsou hodnoty d_j v intervalu $(-\infty, 0)$ jedná se o nulovou preferenci, tj. hodnota $P(d) = 0$. V intervalu $(0, p)$ jsou hodnoty $P(d)$ rovny podílu d a p a od p výše se pak hodnoty $P(d)$ rovnají číslu jedna.

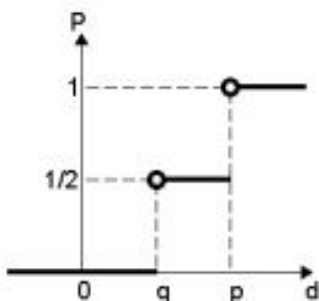
$$P(d) = \begin{cases} 0, & d \leq 0 \\ \frac{d}{p}, & 0 < d \leq p \\ 1, & d > p \end{cases}$$



Obrázek 3.3: Průběh preferenční funkce č. 3

- Preferenční funkce č. 4: úroňňová

Úroňňovou funkci kritéria je vhodné použít v případě, kdy není potřeba znát dokonalou informaci o síle preference. Stačí pouze vědět, zda se jedná o absolutní preference, poloviční preference či indiferenci. Parametr q ohraničuje indifferenční oblast a parametr p udává šíři oblasti pro poloviční preference.

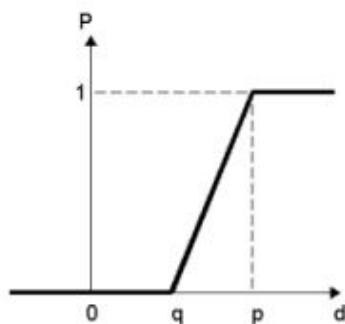


Obrázek 3.4: Průběh preferenční funkce č. 4

$$P(d) = \begin{cases} 0, & d \leq q \\ \frac{1}{2}, & q < d \leq p \\ 1, & d > p \end{cases}$$

- Preferenční funkce č. 5: lineární s prahem indiference

Lineární funkce s prahem indiference je podobná lineární funkci, liší se pouze přidáním indifferenční oblasti.

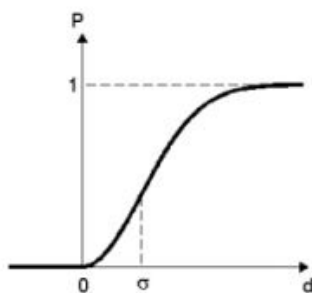


Obrázek 3.5: Průběh preferenční funkce č. 5

$$P(d) = \begin{cases} 0, & d \leq q \\ \frac{d-q}{p-q}, & q < d \leq p \\ 1, & d > p \end{cases}$$

- Preferenční funkce č. 6: gaussova funkce

Gaussovo funkce má pouze jeden parametr σ . Jakýkoliv malý rozdíl d_j znamená preferenci.



Obrázek 3.6: Průběh preferenční funkce č. 6

$$P(d) = \begin{cases} 0, & d \leq 0 \\ 1 - e^{-\frac{d^2}{2\sigma^2}}, & d > 0 \end{cases}$$

3.3.6. Agregované preference

Pokud jsou známy dílčí preference varianty x_i vůči variantě x_z (a naopak) podle všech kritérií a jsou zadány váhy kritérií, spočítá se agregovaná preference

x_i vůči x_z (a naopak), $i, z = 1, 2, 3, \dots, n$, podle vzorce:

$$\begin{cases} \pi(x_i, x_z) = \sum_{j=1}^m P_j(x_i, x_z)v_j & x_i, x_z \in X \\ \pi(x_z, x_i) = \sum_{j=1}^m P_j(x_z, x_i)v_j & x_i, x_z \in X \end{cases}$$

Z preferencí lze vytvořit matici vzájemných agregovaných preferencí, 3.5.

	x_1	x_2	\dots	x_i	\dots	x_n
x_1	0	$\pi(x_1, x_2)$	\dots	$\pi(x_1, x_i)$	\dots	$\pi(x_1, x_n)$
x_2	$\pi(x_2, x_1)$	0	\dots	$\pi(x_2, x_i)$	\dots	$\pi(x_2, x_n)$
\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
x_i	$\pi(x_i, x_1)$	$\pi(x_i, x_2)$	\dots	0	\dots	$\pi(x_i, x_n)$
\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
x_n	$\pi(x_n, x_1)$	$\pi(x_n, x_2)$	\dots	$\pi(x_n, x_i)$	\dots	0

Tabulka 3.5: Matice agregovaných preferencí

3.3.7. Toky ohodnocení

Jestliže se každá alternativa x_i porovnává se všemi ostatními uvažovanými alternativami, kterých je $(n-1)$, je možné definovat kladné a záporné toky pomocí matice agregovaných preferencí.

- Kladný tok ϕ^+ představuje průměrnou sílu preference varianty x_i před ostatními. Z tabulky 3.5 lze kladný tok pro variantu x_i vypočítat jako:

$$\phi^+(x_i) = \frac{1}{n-1} \sum_{x \in X} \pi(x_i, x),$$

$$i = 1, 2, 3, \dots, n.$$

- Záporný tok ϕ^- představuje, jak jsou ostatní varianty v průměru preferované oproti sledované variantě x_i . Z tabulky 3.5 lze záporný tok pro variantu x_i vypočítat jako:

$$\phi^-(x_i) = \frac{1}{n-1} \sum_{x \in X} \pi(x, x_i),$$

$$i = 1, 2, 3, \dots, n.$$

3.3.8. PROMETHEE I

Seřazení variant nastává podle toho, zda nastává mezi variantami preference $(x_i P^I x_z)$, indiference $(x_i I^I x_z)$ či neporovnatelnost $(x_i R^I x_z)$.

Preference varianty x_i před variantou x_z ($x_i P^I x_z$) nastává v případě, jestliže platí některý ze vztahů:

$$\phi^+(x_i) > \phi^+(x_z) \wedge \phi^-(x_i) < \phi^-(x_z) \vee$$

$$\phi^+(x_i) = \phi^+(x_z) \wedge \phi^-(x_i) < \phi^-(x_z) \vee$$

$$\phi^+(x_i) > \phi^+(x_z) \wedge \phi^-(x_i) = \phi^-(x_z).$$

K indiferenci variant dochází jestliže:

$$\phi^+(x_i) = \phi^+(x_z) \wedge \phi^-(x_i) = \phi^-(x_z).$$

K neporovnatelnosti variant dochází jestliže:

$$\phi^+(x_i) > \phi^+(x_z) \wedge \phi^-(x_i) > \phi^-(x_z) \vee$$

$$\phi^+(x_i) < \phi^+(x_z) \wedge \phi^-(x_i) < \phi^-(x_z).$$

K neporovnatelnosti variant dochází zpravidla tehdy, pokud je jedna varianta silná na určité skupině kritérií, kde druhá varianta zaostává, a naopak existují-li kritéria, kde je první varianta slabá a druhá silná. V takových případech metoda PROMETHEE I nedoporučuje rozhodovateli žádnou variantu jako lepší.

3.3.9. PROMETHEE II

Metoda PROMETHEE II je založena pouze na čistých tocích a definuje pouze relace ostré preference $(x_i P^{II} x_z)$ a indiference $(x_i I^{II} x_z)$, to znamená, že oproti hodnocení PROMETHEE I zde neexistuje situace, že by dvě varianty byly nesrovnatelné. Metoda již porovnává a řadí varianty kompletně, čehož dosahuje prostřednictvím čistých toků:

$$\phi(x_i) = \phi^+(x_i) - \phi^-(x_i),$$

$i = 1, 2, 3, \dots, n$. Pro hodnoty $\phi(x_i)$ platí:

$$\phi(x_i) > \phi(x_z) \Leftrightarrow x_i P^{II} x_z$$

$$\phi(x_i) = \phi(x_z) \Leftrightarrow x_i I^{II} x_z,$$

kde $x_i P^{II} x_z$ znamená, že mezi variantami x_i a x_z nastává preference a $x_i I^{II} x_z$ znamená, že mezi variantami nastává indiference.

V praxi se doporučuje používat obě metody (PROMETHEE I i PROMETHEE II) současně, protože ačkoliv se metodou PROMETHEE II dosáhne kompletního pořadí, může analýza nesrovnatelnosti (PROMETHEE I) pomoci při určení správného konečného pořadí.

3.4. Metoda vícekriteriální funkce utility

Kapitola a její podkapitoly jsou vypracovány podle zdrojů [1], [14], [21] a [22].

Vícekriteriální funkce utility je někdy také označována jako funkce užitku, užitková funkce, funkce hodnoty či preferenční funkce. Cílem užitkové funkce je číselně vyjádřit preference.

3.4.1. Konstrukce vícekriteriální funkce utility

Pokud platí předpoklad preferenční nezávislosti kritérií, pak lze celkovou hodnotící funkci (kardinální funkci utility) vyjádřit v aditivním tvaru:

$$u(x) = \sum_{j=1}^m v_j u_j(x),$$

$$v_j \geq 0, j = 1, 2, 3, \dots, m, \sum_{j=1}^m v_j = 1.$$

Konstrukce aditivní vícekriteriální funkce utility lze rozdělit do třech kroků:

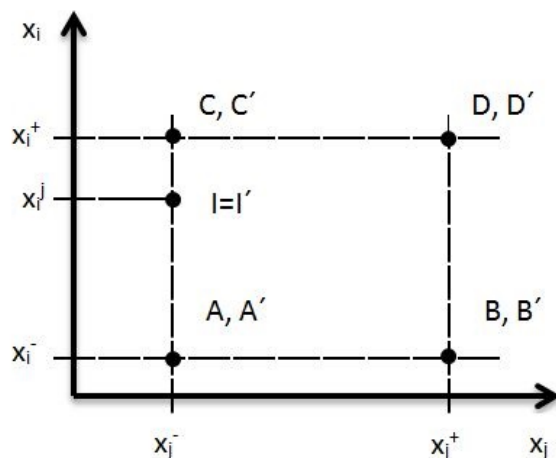
1. ověření preferenční nezávislosti kritérií,
2. konstrukce dílčích funkcí utility pro každé kritérium $u_j, j = 1, 2, 3, \dots, m$, pomocí metody dělicích bodů,
3. stanovení vah kritérií $v_j, j = 1, 2, 3, \dots, m$, pomocí kompenzační metody.

3.4.2. Ověření preferenční nezávislosti kritérií

Definice 3.4.1 ([14])

Nechť $K = \{K_1, \dots, K_m\}$ je množina kritérií, $m > 2$. Řekneme, že dvojice kritérií $K_i, K_j, i \neq j, i, j \in \{1, 2, \dots, m\}$ je preferenčně nezávislá na ostatních kritériích, jestliže preferenční pořadí variant při měnících se hodnotách kritérií K_i, K_j a pevných hodnotách ostatních kritérií nezávisí na těchto pevně zvolených hodnotách. Kritéria z množiny $K = \{K_1, \dots, K_m\}$ nazýváme preferenčně nezávislá, jestliže každá dvojice kritérií z této množiny je preferenčně nezávislá na ostatních kritériích.

Předpokládají se kritéria K_i, K_j s rostoucí preferencí (pro kritéria s klesající preferencí se postupuje analogicky). Praktický postup ověřování preferenční nezávislosti dvojice kritérií je popsán v následujících krocích. Pro každé kritérium $K_j, j = 1, 2, 3, \dots, m$, se označí x_j^+ jako nejlepší a x_j^- jako nejhorší hodnota, které kritérium dosahuje na dané množině variant X .



Obrázek 3.7: Ověření preferenční nezávislosti

- Graficky se zobrazí dvojice nejhorších a nejlepších důsledků variant vzhledem ke kritériím K_i, K_j . Díky těmto hodnotám v grafu vzniknou čtyři body A, B, C, D .

U ostatních hodnot kritérií se předpokládají nejhorší hodnoty, které jsou na dané množině variant dosaženy.

- Rozhodovatel zjistí, zda je více preferován nárůst z nejhoršího důsledku x_j^- na nejlepší x_j^+ vzhledem ke kritériu K_j , což tedy znamená posun z bodu A do B . V této situaci platí $B \succ C$. Druhou možností je to, že je více preferován nárůst z nejhoršího důsledku x_i^- do nejlepšího x_i^+ vzhledem ke kritérium K_i , tedy posun z bodu A do C , tedy $C \succ B$. Dále se předpokládá $C \succ B$.
- Na úsečce AC pak lze nalézt bod I , který je stejně hodnocený, jako bod B .
- Nyní se předpokládají nejlepší hodnoty ostatních kritérií. Např.

$$A' = (x_1^+, \dots, x_i^-, \dots, x_j^-, \dots, x_m^+),$$

$$B' = (x_1^+, \dots, x_i^-, \dots, x_j^+, \dots, x_m^+).$$

Zkoumá se zda $C' \succ B'$:

- jestliže ano \Rightarrow zkoumá se zda $I \sim I'$:
 - * pokud ne – K_i, K_j jsou závislá na ostatních,
 - * pokud ano – K_i, K_j jsou nezávislá na ostatních,
- jestliže ne $\Rightarrow K_i, K_j$ jsou závislá na ostatních.

3.4.3. Konstrukce dílčích funkcí utility

Dílčí funkce utility jsou vytvořeny pomocí metody dělících bodů.

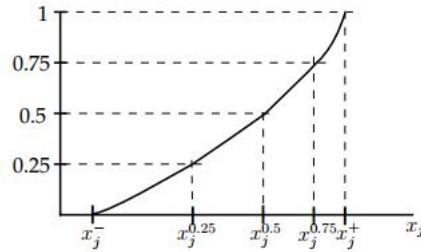
Předpokládá se kvantitativní kritérium K_j s rostoucí preferencí. Předpokládá se, že kardinální funkce utility, která se bude konstruovat, bude normována tak, aby:

$$u_j(x_j^-) = 0,$$

$$u_j(x_j^+) = 1.$$

Rozhodovatel stanoví hodnotu kritéria $x_j^{0.5}$, tak že zvýšení hodnoty kritéria z x_j^0 na hodnotu $x_j^{0.5}$ si cení stejně jako nárůstu hodnoty z $x_j^{0.5}$ na x_j^1 , tzn. $u_j(x_j^{0.5}) =$

0.5. Hodnota kritéria $x_j^{0.25}$ se stanoví tak, že zvýšení hodnoty kritéria z x_j^0 na hodnotu $x_j^{0.25}$ si rozhodovatel cení stejně jako nárůstu hodnoty z $x_j^{0.25}$ na $x_j^{0.5}$, tzn. $u_j(x_j^{0.25}) = 0.25$. Analogicky se postupuje při sestrojení bodu $x_j^{0.75}$, pro který platí $u_j(x_j^{0.75}) = 0.75$. Takto zvolenými body se proloží dílčí funkce utility, obrázek 3.8.



Obrázek 3.8: Proložení funkce dělícími body

Při praktických výpočtech, pokud je chování kardinální funkce utility na jejím definičním oboru poměrně jednoduché, např. funkce utility je rostoucí a konkávní, lze příslušnou kardinální funkci utility aproximovat mocninnou funkcí:

$$y = u_j(x_j) = \left(\frac{x_j - x_j^-}{x_j^+ - x_j^-} \right)^a.$$

Pokud je pro funkci u_j vedle bodů $(x_j^-, 0)$ a $(x_j^+, 1)$ zadán pouze jeden další bod $(x_j^{0.5}, 0.5)$, pak lze a přímo vypočítat pomocí vztahu:

$$0.5 = u_j(x_j^{0.5}) = \left(\frac{x_j^{0.5} - x_j^-}{x_j^+ - x_j^-} \right)^a$$

tj.

$$a = \frac{\ln(0.5)}{\ln \frac{x_j^{0.5} - x_j^-}{x_j^+ - x_j^-}}.$$

Pokud $a = 1 \Rightarrow u$ je lineární, $a > 1 \Rightarrow u$ je konvexní, $a < 1 \Rightarrow u$ je konkávní.

3.4.4. Stanovení vah dílčích utilít kompenzační metodou

Předpokládá se konečná množina variant X . Předpokládá se, že u kritéria K_i se hodnotí nárůst, z nejhorší dosažené hodnoty x_i^- na nejlepší dosaženou hodnotu

x_i^+ , jako nejvýznamnější v porovnání se všemi ostatními kritérii. Předpokládá se, že pro kritéria K_j , $j = 1, 2, 3, \dots, m$, byly sestrojeny normované kardinální dílčí funkce utility u_j . Vícekriteriální funkce utility je tedy ve tvaru:

$$u(x) = \sum_{j=1}^m v_j u_j(x).$$

Označí se (x_j^+, x_j^{-0}) jako varianta, která nabývá u kritéria K_j nejlepší hodnoty a nejhorsích hodnot u všech ostatních kritérií. Z toho, jak bylo vybráno kritérium K_i , plyne, že pro libovolné $j \in \{1, 2, 3, \dots, m\}$, $j \neq i$, platí

$$u(x_i^+, x_i^{-0}) \geq u(x_j^+, x_j^{-0}).$$

Hodnota x_i^j kritéria K_i se stanoví tak, aby varianty (x_i^j, x_i^{-0}) a (x_j^+, x_j^{-0}) byly indiferentní (zápis (x_i^j, x_i^{-0}) značí alternativu, jejíž hodnota u kritéria K_i nabývá hodnoty x_i^j a u všech ostatních kritérií nabývá nejhorsích hodnot), tedy aby platilo:

$$u(x_i^j, x_i^{-0}) = u(x_j^+, x_j^{-0}).$$

Znamená to, že pokud hodnota j -tého kritéria poklesne z x_j^+ na x_j^- , lze tento pokles kompenzovat nárůstem i -tého kritéria z x_i^- na x_i^j .

Dílčí funkce utility jsou normované, tj. $u_j(x_j^-) = 0$ a $u_j(x_j^+) = 1$, pro $j = 1, 2, 3, \dots, m$, proto po dosazení variant (x_i^j, x_i^{-0}) a (x_j^+, x_j^{-0}) do agregačního vzorce $u(x) = \sum_{j=1}^m v_j u_j(x)$ vznikne:

$$u(x_i^j, x_i^{-0}) = v_i u_i(x_i^j),$$

$$u(x_j^+, x_j^{-0}) = v_j.$$

Z rovnosti $u(x_i^j, x_i^{-0}) = u(x_j^+, x_j^{-0})$ celkových utilit obou variant pro podíl normovaných vah platí:

$$\frac{v_j}{v_i} = u_i(x_i^j).$$

Pro nenormované váhy je hodnota podílu stejná; zvolí-li se $w_i = 1$, pak:

$$w_j = \frac{v_j}{v_i} = u_i(x_i^j)$$

pro $j = 1, 2, 3, \dots, m$, $j \neq i$. Pro získání výsledných vah je nutné váhy w_j , $j = 1, 2, 3, \dots, m$, znormovat:

$$v_j = \frac{w_j}{\sum_{j=1}^m w_j}.$$

3.4.5. Celková funkce utility v aditivním tvaru

Předpokládají se kritéria K_j , pro $j = 1, 2, 3, \dots, m$. Pro každou dvojici kritérií platí, že je preferenčně nezávislá na ostatních. Předpokládá se, že $u_j(x_j)$, pro $j = 1, 2, 3, \dots, m$, jsou dílčí kardinální funkce utility stanovené metodou dělicích bodů a předpokládají se váhy $v_j \geq 0$, pro $j = 1, 2, 3, \dots, m$, kritérií stanovené výše popsanou kompenzační metodou. Pokud jsou všechny předpoklady splněny, je možno vyjádřit celkovou užitkovou funkci v aditivním tvaru:

$$u(x) = \sum_{j=1}^m v_j u_j(x_j),$$

$$v_j \geq 0, j = 1, 2, 3, \dots, m, \sum_{j=1}^m v_j = 1.$$

Kapitola 4

Představení vybraných systémů

Tři vybrané softwary, které používají různé metody vícekriteriálního rozhodování, jsou v této kapitole popsány podrobněji. Pomocí programů D-Sight a Make It Rational a 1000Minds je vyřešen rozhodovací problém výběr automobilu.

Rozhodovací problém: Mladá rodina vybírá automobil. Měl by splňovat následující požadavky – nízkou spotřebu, velký zavazadlový prostor. Maximální možnou cenu rodina zvolila půl milionu. Pokud by byla cena nižší než čtyři sta tisíc bude ji rodina považovat za dostatečně nízkou, do pěti set tisíc za vysokou, ale i tak pro rodinu možnou.

Rodina se rozhoduje mezi alternativami v tabulce 4.1.

Alternativy	Cena (v 1000 Kč)	Zavazadlový prostor (v l)	Spotřeba (v l/100 km)
Citroën Grand C4 Picasso	450	625	4.2
Dacia Dokker 1.2	350	800	5.8
Opel Combo Tour 1.4	400	510	6.3

Tabulka 4.1: Alternativy

4.1. D-Sight

Tato kapitola a její podkapitoly jsou popsány ze zdroje [6].

Quantin Hayez, výzkumník v Computer & Decision Engineering Department na Univerzitě Libre v Bruselu, se během studií rozhodl hlouběji prozkoumat metodu vícekritériálního rozhodování – PROMETHEE. Na konci roku 2007 se vytvořil na základě této metody software. Tento program byl úspěšný a v roce 2011 se z Bruselské univerzity rozšířil na volný trh.

Software je používán jak pro výzkum, tak pro výuku. Instituce, které D-Sight používají jsou například: Korea University Business School (Korea), KU Leuven (Belgie), University of Architecture, Civil Engineering and Geodesy of Sofia (Bulharsko), Université du Québec á Chicoutimi (Kanada), Friedrich-Schiller-University Jean (Německo), The Hong Kong Polytechnic University (Čína), University of Delaware (Delaware - USA), University of Rijeka (Chorvatsko), Italferr (Itálie), Public Health Agency of Canada (Kanada), Unesco-IHE (Nizozemsko), Engees (Francie), DCNS Group (Francie), Brussels Airport Company (Belgie), German Aerospace Center (Německo), Unesco-IHE (Nizozemsko), STIB (Belgie), Université Jean Monnet (Francie), Fluxys (Belgie), University of Saskatchewan (Kanada), Politechnika Warszawska (Polsko), University of Guelph (Kanada), University of Surrey (Velká Británie), University of Pretoria (Jihoafrická republika).

V dnešní době software nabízí řešit rozhodovací model buď pomocí metody PROMETHEE nebo metodou vícekritériální funkce utility. Neuvádí se, zda program využívá PROMETHEE I či PROMETHEE II.

Software vyžaduje registraci. Zdarma lze využívat po dobu dvou týdnů demo verzi, poté je cena za software D-Sight 300 Euro za rok. U demo verze si uživatel nemůže zvolit jakou metodu chce pro svůj rozhodovací problém využít a automaticky je použita metoda PROMETHEE. Program si sám volí preferenční funkce, prahy preference a indiference pro daná kritéria.

4.1.1. Praktický příklad řešený v D-Sight

Uživatel svůj projekt nejprve pojmenuje a podrobněji popíše, obrázek 4.1.

Create a new project

Výběr rodinného automobilu

Mladá rodina vybírá automobil. Měl by splňovat následující požadavky - nízkou spotřebu, velký zavazadlový prostor, ...

Start Type

Empty project

Start from a template or one of your projects:

Project Example - Cars

Options

Skip the project creation wizard

Cancel [Show related help](#) Create

Or create an example project

Site Selection

This project aims at identifying the best site for setting up a new factory.

Environmental Impact Assessment

This projects aims to assess the environmental impact of different projects in a certain region.

Cars

This project aims at identifying the best car (among a pre-selection of six cars) based on economical, performance and ecological criteria.

Obrázek 4.1: Pojmenování a popis rozhodovacího problému

V dalším kroku uživatel vypíše alternativy, které zvažuje, obrázek 4.2.

Name	
<input type="text"/>	Add
Opel Combo Tour 1.4	Remove
Dacia Dokker 1.2	Remove
Citroën Grand C4 Picasso	Remove

Obrázek 4.2: Tvorba alternativ

Dále je nutné uvést kritéria podle kterých se uživatel rozhoduje, obrázek 4.3.

Name	
<input type="text"/>	<input type="button" value="Add"/>
Zavazadlový prostor (v litrech)	Remove
Cena (v tisících)	Remove
Spotřeba (v l/100 km)	Remove

Obrázek 4.3: Tvorba kritérií

Po vytvoření alternativ a kritérií musí uživatel zvolit váhy jednotlivých kritérií. Software nabízí tři možnosti, jak váhy určit:

- všechny alternativy budou mít stejnou váhu,
- vlastní volba vah,
- párové porovnávání kritérií, které je založeno na Saatyho určování vah, obrázek 4.4.

Spotřeba (v l/100 km) Cena (v tisících)

Spotřeba (v l/100 km) Zavazadlový prostor (v litrech)

Cena (v tisících) Zavazadlový prostor (v litrech)

Obrázek 4.4: Párové porovnávání variant

Jazykový slovník k počtu zbarvených polí:

- jedno zbarvené pole uprostřed obou variant – obě kritéria jsou stejně důležitá

- dvě zbarvená pole nalevo – kritérium nalevo je sotva důležitější než napravo
- tři zbarvená pole nalevo – kritérium nalevo je o něco důležitější než napravo
- čtyři zbarvená pole nalevo – kritérium nalevo je mírně důležitější než napravo
- pět zbarvených polí nalevo – kritérium nalevo je důležitější než napravo
- šest zbarvených polí nalevo – kritérium nalevo je extrémně důležitější než napravo

Po vyhodnocení párového srovnání kritérií software vypočte váhy jednotlivých kritérií, obrázek 4.5.

Name	
Zavazadlový prostor (v litrech)	65.4807 %
Cena (v tisících)	9.5338 %
Spotřeba (v l/100 km)	24.9855 %

Obrázek 4.5: Váhy kritérií

Nutností je také určit povahu kritérií, obrázek 4.6.

Name	
Spotřeba (v litrech/100 km)	minimize ▼
Zavazadlový prostor (v litrech)	maximize ▼
Cena (v tisících)	minimize ▼

Obrázek 4.6: Povaha kritérií

Poté se vyplní číselné hodnoty, kterých alternativy u jednotlivých kritérií nabývají, obrázek 4.7.

	Spotřeba (v litrech/100 km)	Zavazadlový prostor (v litrech)	Cena (v tisících)
Opel Combo Tour 1.4	6.3	510	400
Dacia Dokker 1.2	5.8	800	350
Citroën Grand C4 Picasso	4.2	625	450

Obrázek 4.7: Vyplnění hodnot kritérií u alternativ

Vyhodnocení variant si uživatel může zobrazit jako tabulku nebo sloupcový graf, obrázek 4.8.



Obrázek 4.8: Grafické vyhodnocení variant

Optimální variantou se stal automobil Dacia Dokker 1.2 s celkovým skórem 87.51 z možných 100 bodů.

4.2. 1000Minds

Tato kapitola a její podkapitoly jsou čerpány ze zdroje [2].

Software 1000Minds vyplynul z výzkumu na Univerzitě v Otagu na Novém Zélandu, který řešil metodu, jak upřednostňovat pacienty pro operaci. 1000Minds byl rychle uznán jako nový přístup k vícekritériálnímu i skupinovému rozhodování. V roce 2003 byla založena firma 1000Minds Ltd se sídly v Dunedinu a

Wellingtonu na Novém Zélandu.

Software je používán jak pro výzkum, tak i pro výuku na mnoha univerzitách po celém světě. Instituce, které 1000Minds používají jsou například: Aalborg University (Dánsko), Abacus Bio (Nový Zéland), Alberta Health Services (Kanada), American College of Rheumatology (USA), Argos (Nový Zéland), Arrowrock (USA), Australian National University (Austrálie), Boeing (USA), Bristol City Council (Velká Británie), BVC Investments (Izrael), Centre for Health, Exercise & Sports Medicine (Austrálie), Well Dunedin Health Trust (Nový Zéland), VTT Technical Research Centre (Finsko), South Gloucestershire Council (Nový Zéland), Office of the Currency Comptroller (USA), NZ Institute of Economic Research (Nový Zéland), Northern Cochlear Implant Trust (Nový Zéland), National Health Board (Nový Zéland), UK Department of Health (Velká Británie), Water, Sanitation and Hygiene (mezinárodní), London Borough of Lambeth (Velká Británie), American College of Rheumatology (USA), Colorado State University – Pueblo (USA), Counties Manukau District Health Board (Nový Zéland), Definitive Business Solutions Inc (USA).

Software používá k vyhodnocení nejlepší varianty metodu PAPRIKA. Pro představu, jak 1000Minds pracuje, je vhodné shlédnout demo verzi rozhodovacího problému. Pokud chce uživatel vyhodnotit vlastní problém, je nutné se nejprve do softwaru zaregistrovat. Prvních 21 dní je program zdarma. Po uplynutí zkušební verze je uživateli zaslán mail, ve kterém vývojáři nabízí komunikaci přes Skype. Touto cestou se může uživatel informovat o ceně softwaru či požádat o prodloužení zkušební verze. Cena softwaru 1000Minds je odlišná pro studijní a komerční účely. Univerzitám po celém světě je poskytován software zdarma, pokud na něj nečerpají grant. U komerčních společností je cena individuální.

4.2.1. Praktický příklad řešení v 1000Minds

V prvé řadě software 1000Minds uživatele vyzve k tomu, aby svůj projekt pojmenoval. Poté je nutno zvolit kritéria a jednotlivé kategorie těchto kritérií. Kategorie se řadí od nejméně příznivé po optimální, jak lze vidět na obrázku 4.9.

The image shows a web interface for creating criteria and levels. It consists of two main sections, one for 'Cena (v 1000)' and one for 'Spotřeba (v l/100 km)'. Each section has a text input field for the criterion name, followed by two dropdown menus for 'lowest ranked' and 'highest ranked' levels. There are also buttons for '+ new criterion' and '+ new level'.

Obrázek 4.9: Tvorba kritérií a jejich kategorií

Dále je nutné vypsát alternativy a k nim zvolit odpovídající kategorie, obrázek 4.10.

The image shows a web interface for creating alternatives and categories. It features a 'Name' field with the value 'Dacia Dokker 1.2.'. Below it is a checkbox labeled 'Show level numbers'. There is a table with two columns: 'CRITERIA' and 'LEVEL'. The table has three rows, each with a criterion name and a dropdown menu for the level.

CRITERIA	LEVEL
Cena (v 1000)	Nízká
Spotřeba (v l/100 km)	Vysoká
Zavazadlový prostor (v l)	Velký

Obrázek 4.10: Tvorba alternativ a kategorií

Po vyplnění všech náležitostí vznikne tabulka alternativ, obrázek 4.11.

		CRITERIA		
ALTERNATIVE	Cena (v 1000)	Spotřeba (v l/100 km)	Zavazadlový prostor (v l)	
Citroën Grand C4 Picasso	Vysoká	Nízká	Velký	
Dacia Dokker 1.2.	Nízká	Vysoká	Velký	
Opel Combo Tour 1.4.	Nízká	Vysoká	Malý	

Obrázek 4.11: Tabulka alternativ

Nyní software vytváří dvojice kategorií kritérií a uživatel zvolí jednu dvojici, která je pro něj lepší. Rozhodovatel také může obě považovat za stejně dobré. Tabulku, pro výběr lepší dvojice kategorií, lze vidět na obrázku 4.12.

Kterou z těchto dvou alternativ upřednostňujete?
(given they're identical in all other respects)

Cena (v 1000)
Vysoká

Spotřeba (v l/100 km)
Nízká

this one

this combination is impossible

OR

Cena (v 1000)
Nízká

Spotřeba (v l/100 km)
Vysoká

this one

this combination is impossible

they are equal

[skip this question »](#)

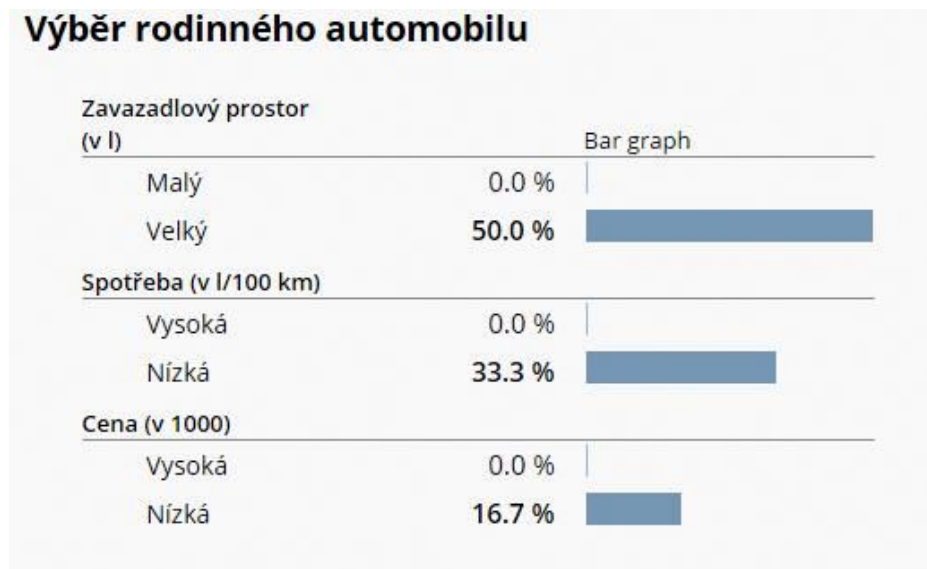
0% complete (0 / 3 potential questions*)

Obrázek 4.12: Výběr lepší dvojice kategorií kritérií

Pokud uživatel pochybuje o tom, že by daná dvojice kategorií kritérií mohla vzniknout, klikne na možnost: this combination is impossible, tedy: tato kombinace není možná a software vyřadí všechny kombinace, jejichž možnost byla

popřena a přetvoří počet porovnávaných dvojic.

Po vyhodnocení všech dvojic kategorií kritérií, které software uživatelé nabídli, program vypočte bodové ohodnocení všech kategorií kritérií, obrázek 4.13.



Obrázek 4.13: Bodové ohodnocení kategorií kritérií

Nyní software vypočte nejlepší alternativu a seřadí všechny alternativy, obrázek 4.14.

ALTERNATIVE	CRITERIA			RANK	TOTAL SCORE
	Cena (v 1000)	Spotřeba (v l/100 km)	Zavazadlový prostor (v l)		
Citroën Grand C4 Picasso	Vysoká	Nízká	Velký	1 st	83.3 %
Dacia Dokker 1.2.	Nízká	Vysoká	Velký	2 nd	66.7 %
Opel Combo Tour 1.4.	Nízká	Vysoká	Malý	3 rd	16.7 %

Obrázek 4.14: Výsledné pořadí alternativ

Software 1000Minds vyhodnotil za nejlepší alternativu automobil Citroën Grand C4 Picasso.

Na obrázku 4.15 lze vidět kódové ohodnocení všech možných alternativ. Např. alternativa 121 (vysoká cena, nízká spotřeba, malý zavazadlový prostor) by se umístila na šestém místě s celkovým bodovým ohodnocením 33.3 %.

Show alternatives with % total scores between and

<input type="checkbox"/>	ALTERNATIVE code hover for words	POSSIBLE RANKS (theoretical range)	RANK out of 8	TOTAL SCORE	% TOTAL SCORE
<input type="checkbox"/>	222	1 to 1 (0 places)	1 st	6	100.0 %
<input type="checkbox"/>	122	2 to 5 (3 places)	2 nd	5	83.3 %
<input type="checkbox"/>	212	2 to 5 (3 places)	3 rd	4	66.7 %
<input type="checkbox"/>	112	4 to 7 (3 places)	4 th =	3	50.0 %
<input type="checkbox"/>	221 ⁺	2 to 5 (3 places)	4 th =	3	50.0 %
<input type="checkbox"/>	121	4 to 7 (3 places)	6 th	2	33.3 %
<input type="checkbox"/>	211 ⁺	4 to 7 (3 places)	7 th	1	16.7 %
<input type="checkbox"/>	111	8 to 8 (0 places)	8 th	0	0.0 %

Obrázek 4.15: Všechny možné alternativy v kódech

4.3. Make It Rational

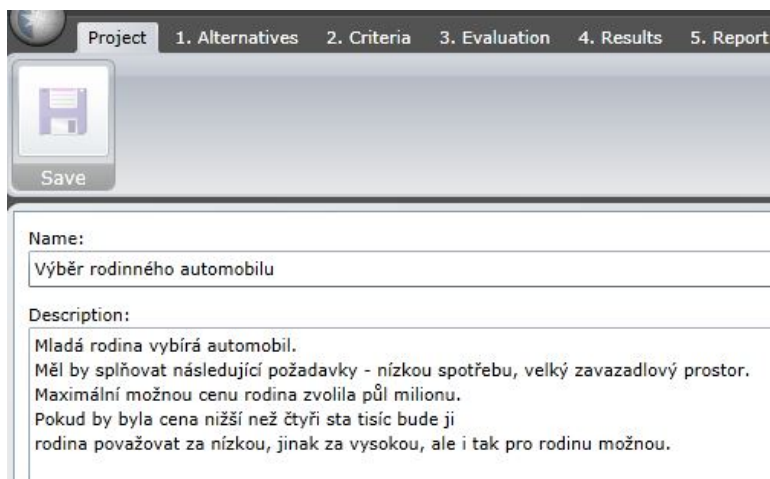
Tato kapitola je popsána ze zdroje [9].

Make It Rational je software, který byl vyvinut v Polsku v Poznani. Software využívá pro hodnocení nejlepší varianty Saatyho Analytický hierarchický proces. Tento program je určen pouze pro komerční nikoli pro akademické účely. Bližší informace o softwaru nejsou na oficiálních stránkách ani volně na webu k dohledání.

Uživatel má možnost využít zdarma demo verzi softwaru nebo zakoupit plnou verzi. Omezení demo verze spočívá v tom, že rozhodovací problém nelze uložit a uživatel se k němu nemůže později vrátit.

4.3.1. Praktický příklad řešený v Make It Rational

Ve volně přístupné demo verzi softwaru Make It Rational si uživatel svůj rozhodovací problém pojmenuje, popřípadě podrobněji popíše, obrázek 4.16.



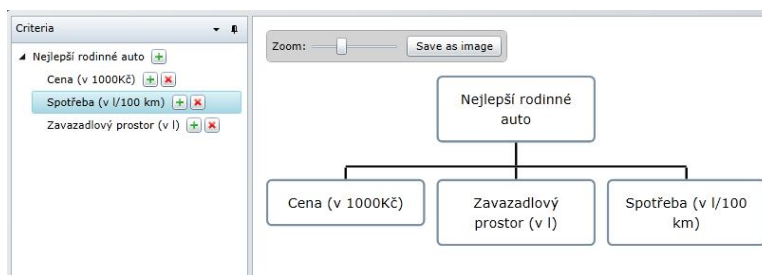
Obrázek 4.16: Pojmenování a popis rozhodovacího problému

Dalším krokem je vyplnění uvažovaných alternativ, obrázek 4.17.



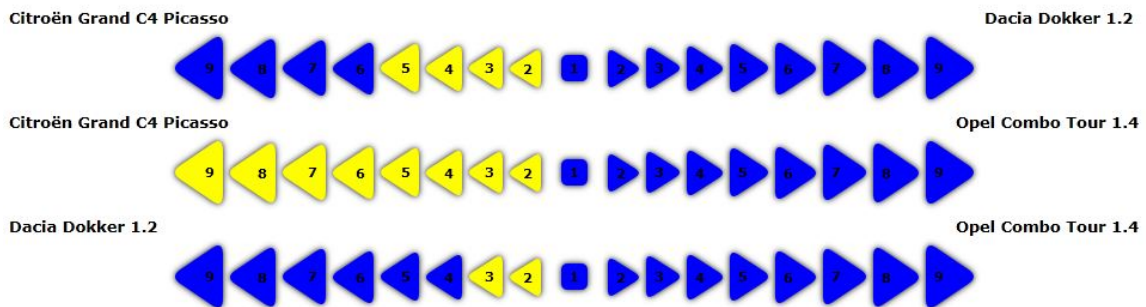
Obrázek 4.17: Tvorba uvažovaných alternativ

Po vytvoření zvažovaných alternativ je nutné napsat cíl projektu a vypsát kritéria, podle kterých se uživatel rozhoduje. Uživatel má také možnost zobrazit si hierarchické schéma problému, obrázek 4.18.



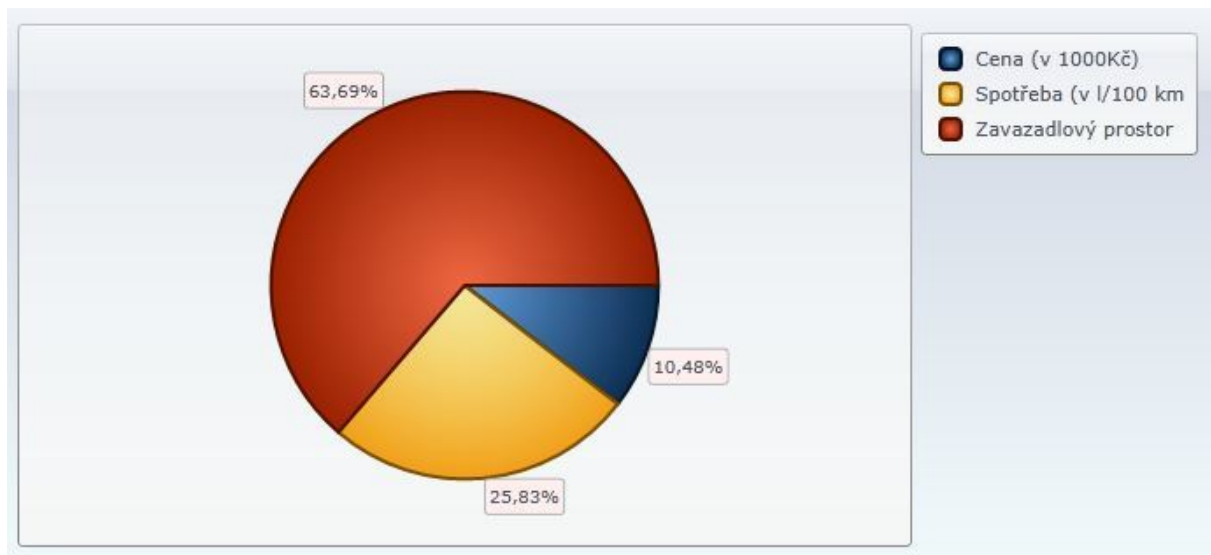
Obrázek 4.18: Tvorba kritérií

Nyní uživatel provádí párové srovnávání kritérií, a to pomocí devítibodové Saatyho stupnice. Poté stejným způsobem hodnotí alternativy v závislosti na jednotlivých kritériích (cena, spotřeba, zavazadlový prostor), obrázek 4.19.



Obrázek 4.19: Porovnávání alternativ

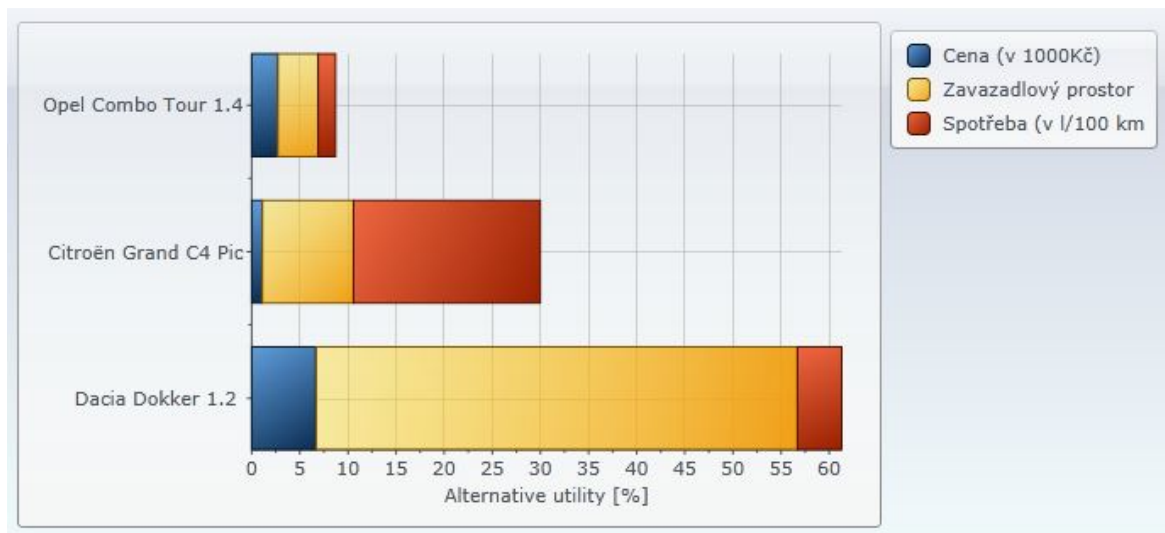
Váhy jednotlivých kritérií lze zobrazit v podobě koláčového grafu, obrázek 4.20.



Obrázek 4.20: Koláčový graf vah kritérií

V dalším kroku se uživateli zobrazí výsledné grafické uspořádání variant, obrázek 4.21.

Software Make It Rational vyhodnotil jako nejlepší variantu automobil Dacia Dokker 1.2.



Obrázek 4.21: Výsledné pořadí variant – graf

I když demo verze neumožňuje uložení rozhodovacího problému, může si uživatel svůj výsledek stáhnout v formě pdf, html verzi nebo jako dokument v McWordu.

Závěr

Hlavním cílem bakalářské práce bylo popsat současnou situaci v oblasti softwarových nástrojů na podporu vícekritériálního hodnocení a rozhodování. Na webových stránkách některých softwarů chyběly základní informace o důvodu vzniku, tvůrcích a zejména bližším popisu užívané metody, což mi bránilo software blíže prozkoumat a popsat. Na druhou stranu, webové stránky jiných softwarů byly velice přehledné a podávaly plnou informaci o programu včetně toho, jaké firmy či univerzity jej využívají nebo kolik stojí roční či měsíční předplatné. Mnoho programů nebylo možno prakticky vyzkoušet, pouze po zaplacení plné verze. Malá skupina programů však nabízela na zkoušku demo verzi nebo dokonce po mailové domluvě poskytla na několik týdnů plnou verzi zdarma.

Dalším cílem bylo popsat a pochopit jedny z nejužívanější a nejznámějších metod rozhodování, na kterých jsou některé zmíněné softwary založeny. S použitím vhodné literatury jsem popsané metody pochopila a podrobně popsala.

Závěrem chci konstatovat, že existuje celá řada nástrojů pro podporu rozhodování. Před využitím některého z nich je tedy vhodné systém důkladněji prozkoumat a zjistit, jak kvalitně dokáže reálný problém popsat. Softwary využívají rozdílné metody, a proto mohou vyhodnotit rozhodovací problém různými způsoby. Podle mého názoru je tedy důležité, aby rozhodovatel využíval software, jehož metodu zná a ví, jak pracuje. Vedle vhodně zvoleného softwaru je základem kvalitního řešení samozřejmě také správná formulace zadání, zejména vhodná volba kritérií rozhodování podle kterých se uživatel rozhoduje.

Tato práce mi přinesla prohloubení mých dosavadních teoretických i praktických znalostí a umožnila mi poznat různá uživatelská prostředí softwarů, které

využívají teorii vícekritériálního rozhodování. Díky prostudovaným systémům pro podporu rozhodování jsem mohla vidět praktické využití teoretických metod a sama jsem si práci v některých softwarech vyzkoušela.

Seznam obrázků

2.1	Uzavřené otázky	15
2.2	Uzavřené otázky	16
3.1	Průběh preferenční funkce č. 1	31
3.2	Průběh preferenční funkce č. 2	32
3.3	Průběh preferenční funkce č. 3	33
3.4	Průběh preferenční funkce č. 4	33
3.5	Průběh preferenční funkce č. 5	34
3.6	Průběh preferenční funkce č. 6	34
3.7	Ověření preferenční nezávislosti	38
3.8	Proložení funkce dělicími body	40
4.1	Pojmenování a popis rozhodovacího problému	45
4.2	Tvorba alternativ	45
4.3	Tvorba kritérií	46
4.4	Párové porovnávání variant	46
4.5	Váhy kritérií	47
4.6	Povaha kritérií	47
4.7	Vyplnění hodnot kritérií u alternativ	48
4.8	Grafické vyhodnocení variant	48
4.9	Tvorba kritérií a jejich kategorií	50
4.10	Tvorba alternativ a kategorií	50
4.11	Tabulka alternativ	51
4.12	Výběr lepší dvojice kategorií kritérií	51
4.13	Bodové ohodnocení kategorií kritérií	52
4.14	Výsledné pořadí alternativ	52
4.15	Všechny možné alternativy v kódech	53
4.16	Pojmenování a popis rozhodovacího problému	54
4.17	Tvorba uvažovaných alternativ	54
4.18	Tvorba kritérií	54
4.19	Porovnávání alternativ	55
4.20	Koláčový graf vah kritérií	55
4.21	Výsledné pořadí variant – graf	56

Seznam tabulek

3.1	Jazykově popsané prvky Saatyho škály	21
3.2	Příklad tvorby Saatyho matice	22
3.3	Indexy $RI(m)$	25
3.4	Výsledná tabulka	27
3.5	Matice agregovaných preferencí	35
4.1	Alternativy	43

Literatura

- [1] Talašová, J.: *Fuzzy metody vícekriteriálního hodnocení a rozhodování*. 1. vyd. Olomouc: Univerzita Palackého, 2003. ISBN 80-244-0614-4.
- [2] 1000Minds [online]. [cit. 2015-10-29]. Dostupné z: <https://www.1000minds.com>.
- [3] Britest [online]. [cit. 2015-01-29]. Dostupné z: <http://www.britest.co.uk>.
- [4] Decisionarium [online]. [cit. 2015-02-02]. Dostupné z: <http://decisionarium.aalto.fi>.
- [5] DEXi: A Program for Multi-Attribute Decision Making. [online]. [cit. 2015-02-20]. Dostupné z: <http://kt.ijs.si/MarkoBohanec/dexi.html>.
- [6] D-SIGHT [online]. [cit. 2015-11-20]. Dostupné z: <http://www.d-sight.com>.
- [7] ElectioVis [online]. [cit. 2015-02-15]. Dostupné z: <https://www.electiovis.com/>.
- [8] M-MACBETH [online]. [cit. 2015-02-16]. Dostupné z: <http://www.m-macbeth.com/en/m-home.html>.
- [9] Make It Rational [online]. [cit. 2015-03-14]. Dostupné z: <http://makeitrational.com>.
- [10] Expert Choice [online]. [cit. 2015-02-16]. Dostupné z: <http://expertchoice.com>.
- [11] PriEsT [online]. [cit. 2015-02-19]. Dostupné z: <https://sourceforge.net/projects/priority/>.
- [12] Ramík, J.: *Vícekriteriální rozhodování - analytický hierarchický proces (AHP)*. 1. vyd. Karviná: Slezská univerzita, 1999. ISBN 80-7248-047-2.
- [13] Jandová, V.: *AHP - její silné a slabé stránky*. [online]. Olomouc, 2012 [cit. 2015-03-25]. Dostupné z: <http://library.upol.cz/ar1-upol/cs/>. Univerzita Palackého v Olomouci.

- [14] Fotr, J. a Píšek, M.: *Exaktní metody ekonomického rozhodování*. 1. vyd. Praha: Academia, 1986. Studie ČSAV.
- [15] Turčíněk, P.: *Příspěvek k řešení metodiky sledování a hodnocení*. Brno, 2008. Mendelova zemědělská a lesnická univerzita v Brně.
- [16] Koubová, K.: *Vícekriteriální hodnocení variant za jistoty – metody rozhodování založené na párovém srovnávání variant*. Pardubice, 2008. Univerzita Pardubice.
- [17] Pokorný, T.: *Grafické řešení úloh vícekriteriálního hodnocení variant*. [online]. Praha, 2014 [cit. 2015-03-27]. Dostupné z: <https://www.vse.cz/vskp/>. Vysoká škola ekonomická v Praze.
- [18] Procházková, L.: *Uplatnění metod třídy PROMETHEE v praktických analýzách*. [online]. Praha, 2014 [cit. 2015-04-18]. Dostupné z: <https://www.vse.cz/vskp/>. Vysoká škola ekonomická v Praze.
- [19] Brožová, H., Houška, M. a Šubrt, T.: *Modely pro vícekriteriální rozhodování*. 1. vyd. Praha: Credit, 2003. ISBN 80-213-1019-7.
- [20] Licek, T.: *Vícekriteriální rozhodování agentů v ekonomických prostředích* [online]. Hradec Králové, 2015 [cit. 2015-04-01]. Dostupné z: <https://theses.cz>. Univerzita Hradec Králové.
- [21] Zemková, B.: *Využití fuzzy množin v oblasti řízení lidských zdrojů*. [online]. Olomouc, 2012 [cit. 2015-02-20]. Dostupné z: <https://theses.cz>. Univerzita Palackého v Olomouci.
- [22] Friebešová, J.: *Teorie užitku*. [online]. nedatováno, studijní opora, [cit.2016-03-05]. Dostupné z: <http://www2.ef.jcu.cz/jfrieb/>.