Poděkování

Rád bych tímto poděkoval vedoucí mé diplomové práce RNDr. Vlastě Kaňkové, CSc. za cenné rady, návrhy a připomínky k této práci a za čas strávený při konzultacích. V neposlední řadě děkuji i za poskytnutí potřebné literatury.

Dále bych chtěl poděkovat svým rodičům za podporu při studiu.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 15. prosince 2005 Jan Líkař

Obsah

Název práce: Vícekriteriální optimalizační úlohy s náhodným elementem a stochastické programování
Autor: Jan Líkař
Katedra: Katedra pravděpodobnosti a matematické statistiky
Vedoucí diplomové práce: RNDr. Vlasta Kaňková, CSc., ÚTIA AV ČR
E-mail vedoucího: kankova@utia.cas.cz

Abstrakt: V praxi jsme často nuceni řešit úlohy, ve kterých posuzujeme současně více protichůdných kriterií. Nazýváme je úlohami vícekriteriální optimalizace. V diplomové práci se zabýváme právě takovými úlohami a především rozlišujeme, zda závisí na náhodných parametrech. Pokud nezávisí, mluvíme o deterministických úlohách. Pokud však na náhodě závisí, jedná se o úlohy stochastické. V deterministické části si ukazujeme, jakými způsoby můžeme získat "nejlepší" řešení dané úlohy a popisujeme vlastnosti takových řešení. Ve stochastickém případě úlohu převádíme na příslušnou deterministickou, kterou již umíme řešit. Představujeme různé způsoby transformací. Pro názornost si uvedené pojmy a postupy demonstrujeme na příkladech. Jako numerickou ilustraci zpracováváme optimální složení portfolia, kde využíváme reálná data.

Klíčová slova: vícekriteriální deterministická úloha, (vlastní) eficientní řešení, vícekriteriální stochastická úloha

Title: Multiobjective Optimization Problems with Random Element and Stochastic Programming
Author: Jan Líkař
Department: Department of Probability and Mathematical Statistics
Supervisor: RNDr. Vlasta Kaňková, CSc., UTIA AS CR
Supervisor's e-mail address: kankova@utia.cas.cz

Abstract: In practice we often have to solve optimization problems with several criteria. These problems are called multicriteria optimization problems. Such problems are presented in this thesis. It is important, whether parameters take unknown values at the moment of making decision. If these parameters are random variables, resulting problem is called stochastic multiobjective problem, otherwise it is called deterministic multiobjective problem. We describe how to choose some "good" solutions of deterministic problem. We investigate their relations as well. In the stochastic case we have to convert such problem to deterministic one. We introduce some possibilities how to do it. Then we are able to solve the problem. These concepts are demonstrated using examples. We present a numerical illustration as well (the Portfolio Selection problem).

Keywords: deterministic multiobjective problem, (properly) efficient solution, stochastic multiobjective problem

Kapitola 1

Úvod

1.1 Úvodní náhled

Ačkoliv si to možná ani příliš neuvědomujeme, v každodenním životě se sami setkáváme s nejrůznějšími jednoduchými optimalizačními úlohami, které jsme nuceni řešit.

Chceme-li se ráno dostat do práce, přemýšlíme o nejvhodnějším způsobu. Upřednostníme rychlejší možnost před pomalejší, pohodlnější před méně pohodlnou, levnější před dražší. Někdy musíme zaplatit složenku, proto si naplánujeme cestu kolem pošty. Zamyslíme se nad všemi těmito okolnostmi a vybereme si pro nás nejlepší variantu.

Při vybírání jídla v restauraci, volbě náplně našeho volného času, nakupování a mnoha jiných činnostech se chováme stejně. Snažíme se vždy něco maximalizovat (nebo minimalizovat) tak, abychom z dané situace vytěžili co nejvíce.

Za tímto účelem se vyvinula teorie optimalizace. Tento vědní obor řeší mnohem složitější problémy, ovšem základní myšlenka je stejná. Jeho snahou je nalézt optimální řešení zadané úlohy v závislosti na různých omezeních.

Jako velmi známý příklad pro ilustraci lze uvést následující maximalizační úlohu lineárního programování

$$\max \sum_{j=1}^{n} c_j x_j$$

za podmínek

$$\begin{aligned} \sum_{j=1}^{n} a_{ij} x_j &= b_i, \quad i = 1, \ldots, m, \\ x_j &\geq 0 \quad \forall j. \end{aligned}$$

Vektorově úlohu zapíšeme

$$\max\ c^T x \quad \text{za podmínek } Ax = b,\ x \geq 0.$$

$A(m,n)$, $b(m,1)$, $c(1,n)$ jsou dané matice.

Jde nám o nalezení optimálního řešení x^{opt}, případně o množinu přípustných řešení $X = \{x \mid Ax = b,\ x \geq 0\}$.

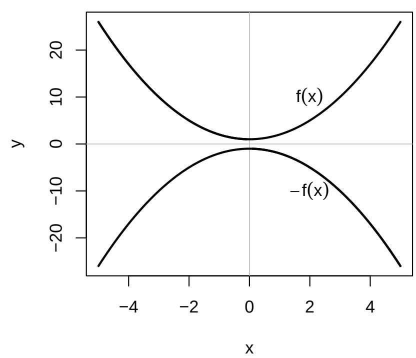
Pokud optimální řešení x^{opt} existuje, pak náleží do množiny přípustných řešení X a platí $c^T x^{opt} \geq c^T x \quad \forall x \in X$.

Optimalizační úlohy se dělí na dva základní typy – deterministické a stochastické. Výše uvedený příklad je klasickou ukázkou deterministické úlohy, neboť všechny parametry a proměnné jsou známé a předem dané. S jejich pomocí se problém většinou snadno vyřeší. Nefiguruje zde tedy žádná náhoda. Pokud by však naše úloha závisela nějakým způsobem na náhodě, neznali bychom všechny koeficienty předem s určitostí, ale znali bychom jejich rozdělení, jednalo by se o úlohu stochastického programování. Jak však můžeme řešit něco, u čeho neznáme přesné zadání? V takovém případě si pomůžeme tzv. rozhodovacím pravidlem. Jde o to, že původní stochastickou úlohu nahradíme odpovídající deterministickou úlohou. K tomu používáme ono zmiňované rozhodovací pravidlo. Je potřeba znát rozdělení náhodných parametrů, nebo mít nějaké jiné informace o tomto rozdělení (např. z dat naměřených v minulosti). Na náhodě pak deterministická úloha závisí jenom přes odpovídající pravděpodobnostní míru.

Oba tyto typy optimalizačních úloh se dále mohou dělit na jednokriteriální a vícekriteriální. Zde jde o to, zda činíme rozhodnutí pomocí jedné účelové funkce (jednoho kriteria), nebo pomocí více účelových funkcí (více protichůdných kriterií). Když si u stánku kupujeme zmrzlinu, většinou máme na výběr z více stejně drahých druhů. Naším jediným kriteriem je tedy chuť a z matematického hlediska se jedná o úlohu jednokriteriální. Pokud si však půjdeme koupit například automobil, naše volba je složitější. Můžeme si vybrat z mnoha typů, po všech stránkách odlišných. Zohledníme cenu, spotřebu, výkon motoru, značku atd. Zkrátka více protichůdných kriterií. Tady zjevně narážíme na úlohu vícekriteriální.

Poznámka: Jak již bylo zmíněno, v optimalizačních úlohách maximalizujeme, nebo minimalizujeme. Mezi těmito dvěma úkony lze snadno převádět. Úlohu na maximalizaci optimalizované funkce $f(x)$, $x \in X$ převedeme na minimalizační úlohu vztahem

$$\min_{x \in X} f(x) = -\max_{x \in X} (-f(x)).$$



Obrázek 1.1: Ilustrace rovnosti $\min_{x \in X} f(x) = -\max_{x \in X} (-f(x))$.

1.2 Značení

V této práci se používá následující značení:

$c^T \in R^{1\times n}$ ... transpozice vektoru $c \in R^{n\times 1}$

$A^T \in R^{m\times n}$ ... transpozice matice $A \in R^{n\times m}$

$R^n,\ n \geq 1 \quad (R := R^1)$ ... n-rozměrný euklidovský prostor

$R^n_+,\ n \geq 1 \quad (R_+ := R^1_+)$ ... n-rozměrná podmnožina euklidovského prostoru s nezápornými prvky

$B^s,\ s \geq 1 \quad (B := B^1)$ ... borelovská σ-algebra na R^s

$P(R^s)$ ... prostor všech pravděpodobnostních měr na (R^s, B^s)

$\omega = (\omega_1, \ldots, \omega_s),\ \omega : (\Omega, \mathcal{A}, P) \rightarrow (R^s, B^s)$ ... s-rozměrný náhodný vektor definovaný na pravděpodobnostním prostoru $(\Omega, \mathcal{A}, P)$

$\omega_i,\ \omega_i : (\Omega, \mathcal{A}, P) \rightarrow (R, B),\ i = 1, \ldots, s$ ... složky s-rozměrného náhodného vektoru ω definovaného na pravděpodobnostním prostoru $(\Omega, \mathcal{A}, P)$

$E_P(\omega)$ ... střední hodnota vzhledem k rozdělení P náhodného vektoru $\omega \in (\Omega, \mathcal{A}, P)$

$var(\rho)$ ... rozptyl náhodné veličiny ρ definované na pravděpodobnostním prostoru $(\Omega, \mathcal{A}, P)$

$cov(\rho_1, \rho_2)$ ... kovariance náhodných veličin ρ_1, ρ_2 definovaných na pravděpodobnostním prostoru $(\Omega, \mathcal{A}, P)$

x^+ ... kladná část $x \in R^n$

x^- ... záporná část $x \in R^n$

$\overline{Y}$ ... uzávěr množiny $Y \subset R^n$

$\Lambda_0 = \{\lambda \in R^K \mid \lambda = (\lambda_1, \ldots, \lambda_K),\ \lambda_k \geq 0,\ k = 1, \ldots, K,\ \sum\limits_{k=1}^{K} \lambda_k = 1\}$ ... množina všech možných nezáporných vah optimalizovaných funkcí

$\Lambda = \{\lambda \in R^K \mid \lambda = (\lambda_1, \ldots, \lambda_K),\ \lambda_k > 0,\ k = 1, \ldots, K,\ \sum\limits_{k=1}^{K} \lambda_k = 1\}$ ... množina všech možných kladných vah optimalizovaných funkcí

$\|x\|_p = [\sum_{i=1}^{n} |x_i|^p]^{\frac{1}{p}},\ p \geq 1,\ x = (x_1, \ldots, x_n) \quad (\|x\|_\infty = \max_{1 \leq i \leq n} \{|x_i|\})$ ... L_p norma

$\langle a, b \rangle$ ... vektorový součin a a b

$\nabla_x \varphi(x^*)$ ... gradient funkce φ v bodě $x = x^*$

Kapitola 2

Vícekriteriální deterministická optimalizace

V životě se většinou setkáváme s případy, kdy rozhodování je složitější a nezávisí pouze na jediném ukazateli. Kriterií, podle kterých vybíráme "nejvhodnější" řešení našeho problému, bývá často více. Takové úlohy se nazývají úlohy vícekriteriální (jinak též vektorové) optimalizace.

Podle způsobu zadání množiny přípustných variant (řešení) se dělí na dvě základní skupiny:

Vícekriteriální hodnocení variant
Zde máme množinu přípustných variant zadanou předem formou nějakého konečného seznamu. Hodnotíme podle zadané množiny kriterií.

Příklad: Vybírám si v obchodě pračku (z nabízených) podle určitých kriterií (cena, výkon, objem, doba praní).

Vícekriteriální programování
Zde je množina přípustných variant vymezena konečným počtem podmínek, které musí být splněny. Hodnotíme opět podle zadané množiny kriterií.

Příklad: Plán produkce v zemědělství. Tento příklad si v následující části podrobněji rozebereme.

2.1 Příklady vícekriteriálních deterministických úloh

Plán produkce v zemědělství ([10])

Farma pěstující ovoce se rozhodla využít volnou zemědělskou půdu a volný čas svých zaměstnanců na doplňkové pěstování zeleniny. K případnému pěstování bylo vybráno 7 druhů zeleniny $Z_1, \ldots, Z_7$. Potřebná doba na pěstování každého druhu zeleniny v každém měsíci na jeden hektar je udána v tabulce 2.1. V posledním sloupci je uveden volný čas v hodinách zaměstnanců farmy během jednotlivých měsíců.

měsíc	zelenina							volný čas
	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	
leden	100	0	0	20	0	0	48	31594
únor	40	0	0	5	0	4	48	15835
březen	133	48	0	56	0	11	95	27390
duben	14	103	1	63	10	11	15	36718
květen	67	48	1	61	9	0	188	53668
červen	168	77	45	14	8	114	180	0
červenec	60	133	39	240	4	0	0	0
srpen	0	62	11	480	0	0	0	933
září	0	0	153	240	105	0	0	933
říjen	0	0	75	0	55	0	0	0
listopad	10	10	0	2	2	2	10	14075
prosinec	20	0	0	0	0	0	16	13107
celkem	612	481	325	1181	193	142	600	

Tabulka 2.1: Doba potřebná pro pěstování zeleniny.

zelenina	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7
zisk	900	900	900	530	50	150	180

Tabulka 2.2: Zisk z prodeje zeleniny.

Z tabulky 2.1 je patrné, že během měsíců června až října nemají zaměstnanci farmy téměř žádný volný čas. Zelenina ovšem růst nepřestane, proto se musí na toto období najmout brigádníci. Brigádníci vyjdou zaměstnavatele dráž než celoroční zaměstnanci (doprava, strava, ubytování). Předpokládaný zisk z prodeje zeleniny vypěstované na ploše 1 ha máme v tabulce 2.2.

Dále známe tato omezení:

a) k pěstování zeleniny můžeme využívat pozemek s plochou 95 ha,

b) Z_1, Z_2, Z_3 lze pěstovat maximálně na ploše 77 ha,

c) Z_4 lze pěstovat maximálně na ploše 63 ha,

d) Z_6 musíme pěstovat minimálně na ploše 6 ha.

Cíle farmy jsou následující:

A) maximalizovat zisk,

B) minimalizovat celkový čas potřebný na pěstování zeleniny,

C) minimalizovat počet hodin brigádníků,

D) co nejvíce využít volný čas svých zaměstnanců.

Známe-li tedy zadání, můžeme sestavit matematický model této úlohy. Označíme si $x_1, \ldots, x_7$ počet hektarů určený k pěstování zeleniny $Z_1, \ldots, Z_7$. Počet brigádnických hodin v měsících červnu až říjnu označíme $x_8, \ldots, x_{12}$. Potom nám platí následující omezení pro čas potřebný k pěstování jednotlivých druhů zeleniny ve všech měsících:

$$
\begin{aligned}
100x_1 + 20x_4 + 48x_7 &\leq 31594,\\
40x_1 + 5x_4 + 4x_6 + 48x_7 &\leq 15835,\\
133x_1 + 48x_2 + 56x_4 + 11x_6 + 95x_7 &\leq 27390,\\
14x_1 + 103x_2 + x_3 + 63x_4 + 10x_5 + 11x_6 + 15x_7 &\leq 36718,\\
67x_1 + 48x_2 + x_3 + 61x_4 + 9x_5 + 188x_7 &\leq 53668,\\
168x_1 + 77x_2 + 45x_3 + 14x_4 + 8x_5 + 114x_6 + 180x_7 &= x_8,\\
60x_1 + 133x_2 + 39x_3 + 240x_4 + 4x_5 &= x_9,\\
62x_2 + 11x_3 + 480x_4 &\leq 933 + x_{10},\\
153x_3 + 240x_4 + 105x_5 &\leq 933 + x_{11},\\
75x_3 + 55x_5 &= x_{12},\\
10x_1 + 10x_2 + 2x_4 + 2x_5 + 2x_6 + 10x_7 &\leq 14075,\\
20x_1 + 16x_7 &\leq 13107.
\end{aligned}
$$

Uvedené nerovnosti převedeme na rovnosti pomocí proměnných $x_{13}, \ldots, x_{21}$. Tyto proměnné udávají volný čas zaměstnanců farmy. Podmínky a) – d) přepíšeme jako

$$
\begin{aligned}
x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 &= 95,\\
x_1 + x_2 + x_3 &\leq 77,\\
x_4 &\leq 63,\\
x_6 &\geq 6
\end{aligned}
$$

a doplníme do rovnic pomocí proměnných $x_{22}, \ldots, x_{24}$.

Množina přípustných řešení naší úlohy vypadá takto:

$$
\begin{aligned}
X = \{x \in R^{24} \mid\ & 100x_1 + 20x_4 + 48x_7 + x_{13} = 31594,\\
& 40x_1 + 5x_4 + 4x_6 + 48x_7 + x_{14} = 15835,\\
& 133x_1 + 48x_2 + 56x_4 + 11x_6 + 95x_7 + x_{15} = 27390,\\
& 14x_1 + 103x_2 + x_3 + 63x_4 + 10x_5 + 11x_6 + 15x_7 + x_{16} = 36718,\\
& 67x_1 + 48x_2 + x_3 + 61x_4 + 9x_5 + 188x_7 + x_{17} = 53668,\\
& 168x_1 + 77x_2 + 45x_3 + 14x_4 + 8x_5 + 114x_6 + 180x_7 = x_8,\\
& 60x_1 + 133x_2 + 39x_3 + 240x_4 + 4x_5 = x_9,\\
& 62x_2 + 11x_3 + 480x_4 + x_{18} = 933 + x_{10},\\
& 153x_3 + 240x_4 + 105x_5 + x_{19} = 933 + x_{11},\\
& 75x_3 + 55x_5 = x_{12},\\
& 10x_1 + 10x_2 + 2x_4 + 2x_5 + 2x_6 + 10x_7 + x_{20} = 14075,\\
& 20x_1 + 16x_7 + x_{21} = 13107,\\
& x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 = 95,\\
& x_1 + x_2 + x_3 + x_{22} = 77,\\
& x_4 + x_{23} = 63,\\
& x_6 - x_{24} = 6,\\
& x_1, \ldots, x_{24} \geq 0\}.
\end{aligned}
$$

Účelové funkce dostaneme z podmínek A) – D):

- $f_1(x) = 900x_1 + 900x_2 + 900x_3 + 530x_4 + 50x_5 + 150x_6 + 180x_7$
- $f_2(x) = 612x_1 + 481x_2 + 325x_3 + 1181x_4 + 193x_5 + 142x_6 + 600x_7$
- $f_3(x) = \sum_{i=8}^{12} x_i$
- $f_4(x) = \sum_{i=13}^{21} x_i$

Funkci $f_1(x)$ maximalizujeme, zbylé funkce minimalizujeme. Dospěli jsme tedy k úloze vícekriteriálního programování

$$\max_{x \in X} [f_1(x), -f_2(x), -f_3(x), -f_4(x)].$$

Dopravní problém ([10])

Úkolem je co nejlépe sestavit přepravní plán ovoce (na $D+1$ dnů; dny číslujeme $0, \dots, D$) ze zámoří, máme-li k dispozici následující požadavky a omezení. Ovoce se balí ve čtyřech různých místech $B_1, \dots, B_4$ s kapacitou $e_{Bi}(t),\ i = 1, \dots, 4,\ t = 0, \dots, D$. Odtud se zabalené vozí buď rovnou do přístavu (a lodí se hned přepraví přes moře), nebo do některého ze skladů $S_1, \dots, S_3$. Jejich kapacita je $e_{Sj},\ j = 1, \dots, 3$. Sklady se nachází na jiných místech než balírny. Ze skladů se ovoce přepraví do přístavu nějaký další den. Tyto sklady slouží jako rezerva pro zásobování přístavu a také mají velkou výhodou v tom, že obsahují chladící zařízení, díky nimž ovoce vydrží déle čerstvé a po jeho dopravení se tedy snáze prodává. Je logické, že se však za uchovávání ovoce ve skladech platí určitá taxa.

Známe požadavek $d(t),\ t = 0, \dots, D$ na množství ovoce přepravené každý den. Předpokládá se, že v jakýsi nultý den ($t = 0$) se ovoce může balit a chladit, ale ještě se nedováží do přístavu. Proto platí omezení $d(0) = 0$. Další omezení je takové, že v poslední den ($t = D$) se bude ovoce přesouvat pouze do přístavu a nemá cenu nějaké nechávat ve skladech.

Náklady na přepravu 1 kg ovoce z balírny B_i do přístavu jsou n_i. Náklady na přepravu ze skladu S_j do přístavu jsou $\bar{n}_j$ a z balírny B_i do skladu S_j jsou n_{ij}. U všech zmiňovaných míst $i = 1, \dots, 4,\ j = 1, \dots, 3$.

Jak již bylo v úvodu řečeno, hlavním úkolem je sestavit dopravní plán na nejbližších $D+1$ dnů tak, aby byly splněny požadavky na množství dovezeného ovoce ze zámoří, a zároveň aby byla co nejvyšší kvalita dováženého ovoce a naše náklady byly co nejmenší.

Zavedeme proměnné

$o_j^S(t),\quad j = 1, \dots, 3,\ t = 0, \dots, D$ množství ovoce v j-tém skladu na začátku dne t,

$o_i(t),\quad i = 1, \dots, 4,\ t = 0, \dots, D$ množství ovoce poslaného z balírny B_i přímo do přístavu během dne t,

$\bar{o}_j(t), \quad j = 1, \ldots, 3,\ t = 0, \ldots, D$ množství ovoce poslaného ze skladu S_j do přístavu během dne t,

$o_{ij}(t), \quad i = 1, \ldots, 4,\ j = 1, \ldots, 3,\ t = 0, \ldots, D$ množství ovoce poslaného z balírny B_i do skladu S_j během dne t.

Ze zadání odvodíme omezení vzhledem ke kapacitě balíren

$$\sum_{j=1}^{3} o_{ij}(t) + o_i(t) \leq e_{Bi}(t), \quad i = 1, \ldots, 4,\ t = 1, \ldots, D-1,$$

$$\sum_{j=1}^{3} o_{ij}(0) \leq e_{Bi}(0), \quad i = 1, \ldots, 4,$$

$$o_i(D) \leq e_{Bi}(D), \quad i = 1, \ldots, 4.$$

Další omezení se týká požadavku na množství ovoce dodaného každý den do přístavu:

$$\sum_{i=1}^{4} o_i(t) + \sum_{j=1}^{3} \bar{o}_j(t) = d(t), \quad t = 1, \ldots, D.$$

Je logické, že ze skladu nemůžeme odvézt více ovoce, než ho tam je. Tedy

$$\bar{o}_j(t) - o_j^S(t) \leq 0, \quad j = 1, \ldots, 3,\ t = 1, \ldots, D.$$

Vzhledem ke kapacitě skladů platí

$$\sum_{i=1}^{4} o_{ij}(t) + o_j^S(t) \leq e_{Sj}, \quad j = 1, \ldots, 3,\ t = 0, \ldots, D-1.$$

Pro množství ovoce ve skladu S_j na začátku každého dne platí

$$o_j^S(1) = o_j^S(0) + \sum_{i=1}^{4} o_{ij}(0), \quad j = 1, \ldots, 3,$$

$$o_j^S(t) = o_j^S(t-1) + \sum_{i=1}^{4} o_{ij}(t-1) - \bar{o}_j(t-1), \quad j = 1, \ldots, 3,\ t = 2, \ldots, D.$$

Množina přípustných řešení X je tedy množina takových nezáporných $o_j^S(t)$, $o_i(t)$, $\bar{o}_j(t)$ a $o_{ij}(t)$, vyhovující všem podmínkám uvedeným výše.

Úloha bude mít dvě účelové funkce $f_1(x)$ a $f_2(x)$. První z nich popisuje množství ovoce dopravovaného přímo z balíren do přístavu (minimalizujeme). Jinými slovy požadujeme maximální kvalitu ovoce díky chlazení. Druhá účelová funkce popisuje veškeré náklady na přepravu a skladování ovoce. Tu budeme logicky rovněž minimalizovat.

$$f_1(x) = \sum_{t=1}^{D} \sum_{i=1}^{4} o_i(t)$$

$$f_2(x) = \sum_{t=1}^{D}(\sum_{i=1}^{4} n_i o_i(t) + \sum_{j=1}^{3} \bar{n_j}\bar{o_j}(t)) + \sum_{t=0}^{D-1}\sum_{i=1}^{4}\sum_{j=1}^{3} n_{ij} o_{ij}(t)$$

Matematická formulace našeho problému tedy je

$$\min_{x \in X} [f_1(x),\ f_2(x)].$$

2.2 Vícekriteriální hodnocení variant

O vícekriteriálním hodnocení variant se zmíníme jen informativně. Podrobně je tato látka zpracována v knize [7].

Základní pojmy:

Nedominovaná varianta
Předpokládejme, že jednotlivé varianty hodnotíme podle K různých kriterií. Dále předpokládejme, že všechna kriteria jsou maximalizační. Máme dvě varianty $V_a = (v_{a1}, v_{a2}, \ldots, v_{aK})$ a $V_b = (v_{b1}, v_{b2}, \ldots, v_{bK})$. Prvky v_{ai}, resp. v_{bi}, $i = 1, \ldots, K$ odpovídají hodnotám i-tého kriteria pro variantu V_a, resp. V_b. Varianta V_a dominuje variantu V_b, jestliže $V_a \geq V_b$. V obecném případě (pro libovolný počet variant) tedy řekneme, že varianta V^* se nazývá nedominovaná, jestliže v množině rozhodovacích variant neexistuje varianta, která by jí dominovala. Je logické, že nikdy nebudeme vybírat variantu, které jiná varianta dominuje.

Ideální je, pokud je nedominovaná varianta pouze jedna. Ta je pak optimální. S touto situací se však setkáme zřídka. Většinou je nedominovaných variant více, pak volíme kompromisní variantu.

Optimální varianta
Nemá jednoznačnou definici. Je třeba určit, v jakém smyslu má být optimální. Je relativně jednoznačně doporučená k výběru nebo realizaci.

Kompromisní varianta
Reprezentant množiny nedominovaných variant. Nějak vybraná varianta ze všech nedominovaných. Například varianta, která má největší součet určitým způsobem normalizovaných hodnot kriterií. Způsob výběru kompromisní varianty by měl zajišťovat tyto vlastnosti: nedominovanost, determinovanost (aspoň jedna varianta musí být označena jako kompromisní), invariance vzhledem k permutacím kriterií (neboli nezáleží na pořadí kriterií), invariance vzhledem ke změně měřítka hodnot kriterií, nezávislost na identických hodnotách téhož kriteria, invariance vzhledem k přidaným nekompromisním hodnotám, jednoznačnost.

Výhodné je, pokud máme k dispozici informace o preferencích mezi jednotlivými kriterii. Jinými slovy, jak jsou která kriteria důležitá. Vrátíme se k příkladu s výběrem pračky. Pokud je pro nás nejdůležitější cena a naopak na době praní nám až tolik nezáleží, musíme na to při rozhodování brát zřetel. Jak tedy postupovat si načrtneme na třech přístupech.

Aspirační úrovně kriterií
Uživatel si určí tzv. aspirační úrovně kriterií. Jedná se o hodnoty, kterých by měla minimálně dosáhnout hodnocená varianta. Pokud jich dosahuje (a pokud možno přesahuje), jedná se o akceptovatelnou variantu. Jinak jde o neakceptovatelnou variantu. Logicky si nakonec vybereme nějakou z akceptovatelných. Pokud jich je více, můžeme aspirační úrovně zpřísňovat, až nám zbyde poslední – nejvhodnější varianta.

Ordinální informace o kriteriích
Uživatel uspořádá kriteria od nejdůležitějšího po nejméně důležité. Pak se například pokračuje lexikograficky – vyberou se varianty s nejvyšší hodnotou u nejdůležitějšího kriteria, z nich (pokud jich je víc) se vezmou varianty s nejvyšší hodnotou u druhého nejdůležitějšího kriteria atd. U některých konkrétních metod se připouští i existence několika stejně hodnocených kriterií.

Kardinální informace o kriteriích ve formě vah
Důležitost kriterií lze vyjádřit pomocí vektoru vah

$$\lambda = (\lambda_1, \ldots, \lambda_K), \quad \lambda \in \Lambda.$$

Čím je kriterium důležitější, tím mu přiřadíme větší váhu. Uživatel však většinou přesné hodnoty neví, proto existují metody na získání jejich odhadů (metoda pořadí, bodovací metoda, metoda párového srovnání kriterií a další).

2.3 Vícekriteriální programování

2.3.1 Pojmy a vztahy

Deterministickou optimalizační úlohu vícekriteriálního programování můžeme formálně definovat (pro případ maximalizace) následujícím způsobem.

Nechť máme neprázdnou množinu přípustných řešení $X \subset R^n$. Dále nechť máme K reálných účelových funkcí $f_k : R^n \to R,\ k = 1, \ldots, K$.

Hledáme

$$\max_{x \in X} f(x) = \max_{x \in X} [f_1(x), \ldots, f_K(x)]. \tag{2.1}$$

Zcela jistě nejlepší by bylo, pokud by pro úlohu (2.1) existovalo tzv. ideální řešení $x^0 \in X$. Zadefinujeme si jej tímto vztahem:

$$x^0 \in X : f_k(x^0) = \max_{x \in X} f_k(x) \quad \forall k,$$

neboli

$$x^0 \in \bigcap_{k=1}^{K} \arg\max_{x\in X} f_k(x).$$

Jedná se tedy o řešení, které je ve všech směrech (podle všech kriterií) nejlepší. Taková situace však nastává málokdy a vlastně se ani nedá příliš považovat za vícekriteriální problém.

Další pro nás nepříliš zajímavá situace nastane, pokud je jedna z funkcí $f_k(x)$ z nějakého důvodu preferovaná před ostatními – $f_k(x)$ je tzv. dominantní funkce. S dominantní funkcí se setkáme v případech, kdy je tato funkce v řešené úloze velmi důležitá a ostatní jsou poměrně nepodstatné. Pak se problém převede na prostou úlohu

$$\max_{x\in X} f_k(x)$$

a řešení nazveme dominantním řešením úlohy (2.1).

Zadefinujeme si několik pojmů, se kterými se setkáváme ve vícekriteriálním programování. Pro jejich snadnější pochopení si uvedeme nějaké příklady. Na nich je vidět, jak jsou jednotlivé definice užitečné. Následně si zadefinované pojmy přiblížíme popsáním jejich vlastností, omezení, vztahů mezi nimi.

Nejčastěji se pod řešením vícekriteriální optimalizační úlohy (2.1) rozumí nalezení množiny tzv. eficientních řešení.

Definice 2.1: [8] Vektor $x^* \in X$ nazveme eficientním (anglicky *efficient*) řešením vícekriteriální optimalizační úlohy (2.1), jestliže neexistuje vektor $x \in X$ splňující

$$f(x) \geq f(x^*) \text{ a } f(x) \neq f(x^*).$$

Množinu všech eficientních řešení úlohy (2.1) budeme označovat X_E.

Eficientní řešení můžeme vysvětlit tak, že neexistuje jiné přípustné řešení, které by bylo aspoň stejně dobré ve všech kriteriích a podle nejméně jednoho kriteria ostře lepší.

Poznámka: V literatuře se můžeme setkat i s jinými názvy označujícími stejnou věc – nedominované řešení, efektivní řešení, Pareto-optimální řešení aj.

Poznámka: V literatuře se také lze setkat s různými alternativními definicemi eficientního řešení. Jako příklad si uvedeme tzv. nedominované řešení popsané H. Bensonem [2]:

Vektor $x' \in X$ je dominovaný, jestliže existuje nějaký jiný vektor $x \in X$ tak, že $f(x) \geq f(x')$ a aspoň pro jedno $k = 1, \ldots, K$ platí $f_k(x) > f_k(x')$. Řekneme, že vektor $x^* \in X$ je eficientním řešením úlohy (2.1), jestliže není dominovaný žádným jiným vektorem z X.

Definice 2.2: [6] Jestliže $x^* \in X$ je eficientní, potom vektor $f(x^*)$ se nazývá nedominovaný (*nondominated*).
Množinu všech nedominovaných vektorů $y^* = f(x^*) \in Y$, kde $x^* \in X_E$, budeme označovat Y_N.

Zde je potřeba dovysvětlit, že $Y := f(X)$ označuje obraz množiny přípustných řešení X při zobrazení vícesložkovou účelovou funkcí f. Je zřejmé, že v našem případě $Y \subset R^K$. X je vlastně prostor řešení (rozhodnutí) a Y je prostorem kriterií (účelových funkcí). Podle této definice je tedy vektor $y^* \in Y$ nedominovaný, jestliže neexistuje jiný vektor $y \in Y$ takový, že $y^* \leq y$ a $y^* \neq y$.

Poznámka: Pro zjednodušení v některých příkladech zvolíme $y = f(x) = x$ (např. $f(x_1, x_2) = [x_1, x_2]$). V těchto případech tedy platí $X = Y$.

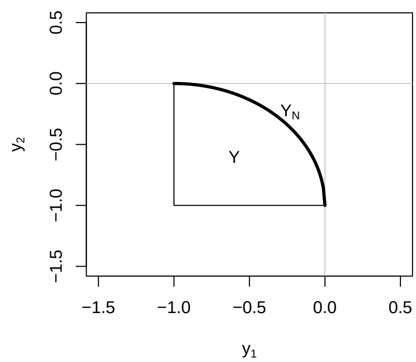
Tento trik se může hodit i při probírání vlastností zadefinovaných pojmů. Díky této rovnosti prostorů řešení (rozhodnutí) a kriterií (účelových funkcí) můžeme popisovat oba prostory X_E a Y_N současně.

Příklad 2.1: [6] Nechť máme přípustnou množinu v prostoru rozhodnutí a kriterií (tedy mějme $X = Y$) zadanou vztahem

$$X = \{(x_1, x_2) \in R^2 \mid (x_1 + 1)^2 + (x_2 + 1)^2 \leq 1,\ -1 \leq x_1 \leq 0,\ -1 \leq x_2 \leq 0\}.$$

Pak tedy

$$Y_N = \{(y_1, y_2) \in Y \mid (y_1 + 1)^2 + (y_2 + 1)^2 = 1\}.$$



Obrázek 2.1: Množiny Y, Y_N.

V první řadě je důležité vědět, zda vůbec nějaká eficientní řešení konkrétní úlohy existují.

Je pochopitelné, že eficientní body (v tomto případě vlastně vektory) musí ležet na hranici množiny X (a nedominované body (vektory) tím pádem na hranici množiny Y). Pokud by tomu tak nebylo, jistě by se v okolí tohoto bodu (mylně považovaného za eficientní) nacházel jiný bod blíže k hranici množiny X (respektive Y). Náš domnělý "eficientní bod" bude dominovaný (v žádném ohledu nebude lepší, v něčem bude dokonce horší) právě tímto bodem a proto se nemůže jednat o eficientní řešení. Eficientní řešení nemůže být dominováno jiným řešením. Z tohoto důvodu logicky musí ležet na hranici množiny X (resp. Y). Za tohoto předpokladu se nabízí následující tvrzení:

Věta 2.1: [6] *Jestliže neprázdná množina Y je otevřená, potom $Y_N = \emptyset$.*

Na druhou stranu se dá očekávat, že platí

Věta 2.2: [6] *Jestliže neprázdná množina Y je kompaktní, potom $Y_N \neq \emptyset$.*

Nyní si uvedeme definici slabě eficientních řešení a porovnáme je s eficientními.

Definice 2.3: [6] Vektor $x^* \in X$ se nazývá slabě eficientní (*weakly efficient*) řešení vícekriteriální optimalizační úlohy (2.1), jestliže neexistuje žádný jiný vektor $x \in X$, pro který platí $f(x) > f(x^*)$.
Množinu všech slabě eficientních řešení úlohy (2.1) budeme označovat X_{WE}.

Definice 2.4: [6] Jestliže $x^* \in X_{WE}$, potom odpovídající vektor $y^* = f(x^*) \in Y$ se nazývá slabě nedominovaný (*weakly nondominated*).
Množinu všech slabě nedominovaných vektorů y^* budeme označovat Y_{WN}.

Příklad 2.2: [6] Uvažujme množinu

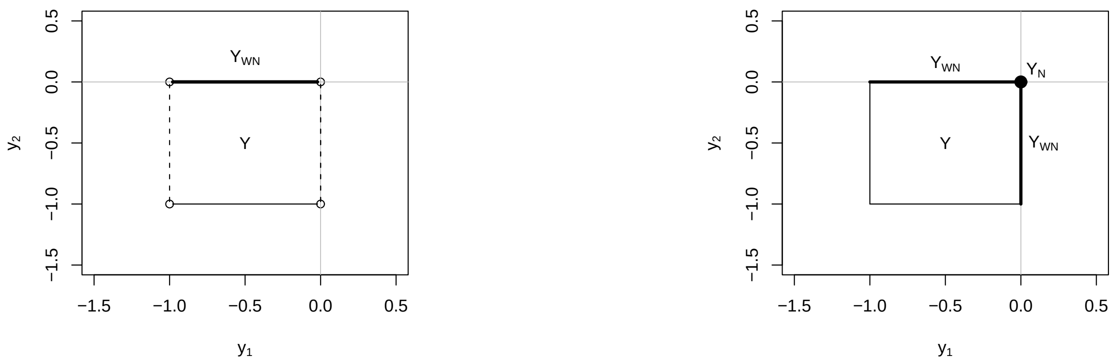
$$Y = \{(y_1, y_2) \in R^2 \mid -1 < y_1 < 0,\ -1 \leq y_2 \leq 0\}.$$

V tomto případě $Y_N = \emptyset$, ale $Y_{WN} = \{y \in Y \mid -1 < y_1 < 0,\ y_2 = 0\}$.

Pokud bychom množinu Y zadali

$$Y = \{(y_1, y_2) \in R^2 \mid -1 \leq y_1 \leq 0,\ -1 \leq y_2 \leq 0\},$$

řešením by bylo $Y_N = \{0\}$ a $Y_{WN} = \{(y_1, y_2) \in Y \mid y_1 = 0 \vee y_2 = 0\}$.



Obrázek 2.2: Množiny Y, Y_N, Y_{WN}.

Nějaké vztahy mezi uvedenými množinami X_E a X_{WE} jsou zřejmé již z definic. Je jasné, že

$$X_E \subset X_{WE} \text{ (tím pádem i } Y_N \subset Y_{WN}).$$

Pokud existuje nějaké eficientní řešení, pak zcela jistě budou existovat i slabě eficientní řešení. Minimálně bude v množině X_{WE} ono existující eficientní řešení. Pro odpovídající nedominovaná (slabě nedominovaná) řešení pochopitelně platí to samé. Na druhou stranu si však nemůžeme myslet, že pokud existuje slabě eficientní řešení, automaticky bude existovat i eficientní řešení. O tom jsme se již přesvědčili v příkladu 2.2. V tomto příkladu je Y_{WN} neprázdná, zatímco Y_N je prázdná.

Jako existenční podmínku pro slabě nedominovaná řešení uvedeme větu 2.2. Jen místo Y_N použijeme Y_{WN}. Tuto existenční podmínku a spojitost funkcí f využijeme k dokázání existence slabě eficientních řešení.

Věta 2.3: [6] *Nechť $X \subset R^n$ je neprázdná a kompaktní. Dále nechť $f : R^n \to R^K$ je spojitá. Potom $X_{WE} \neq \emptyset$.*

Na závěr si přiblížíme vlastní eficientní řešení. Tento pojem zavedl v 60. letech minulého století A. Geoffrion.

Definice 2.5: [8] Vektor $x^* \in X$ nazveme vlastní eficientní (*properly efficient*) řešení vícekriteriální optimalizační úlohy (2.1) právě tehdy, když $x^* \in X_E$ a existuje libovolně velký reálný skalár $M > 0$ takový, že pro každé k a každé $x \in X$ splňující $f_k(x) > f_k(x^*)$ existuje minimálně jedno j takové, že $f_j(x^*) > f_j(x)$ a

$$\frac{f_k(x) - f_k(x^*)}{f_j(x^*) - f_j(x)} \leq M.$$

Množinu všech vlastních eficientních řešení úlohy (2.1) budeme označovat X_{PRE}.

Slovy můžeme tuto definici interpretovat tak, že když u jakéhokoliv kriteria existuje přípustné řešení x lepší než vlastní eficientní řešení x^*, u jiného kriteria se tato ztráta musí zneutralizovat. U každého vlastního eficientního řešení úlohy (2.1), pro každé kriterium, poměr mezi možnou ztrátou (způsobenou tím, že jsme zvolili x^* a ne x; $f_k(x) - f_k(x^*) > 0$) a ziskem $(f_j(x^*) - f_j(x) > 0)$ nemůže být neomezený. Nemůže tedy nastat situace, že $f_j(x^*) - f_j(x) \to 0$ pro každé j takové, že $f_j(x^*) > f_j(x)$.

Poznámka: V literatuře se můžeme setkat i s dalšími českými překlady, jako například ryze eficientní řešení.

Poznámka: Existují i jiné alternativní definice vlastních eficientních řešení. Jejich znění a vztahy mezi nimi si rozebereme později.

Definice 2.6: [6] Jestliže $x^* \in X_{PRE}$, potom odpovídající vektor $y^* = f(x^*) \in Y$ se nazývá vlastní nedominovaný (*properly nondominated*).

Množinu všech vlastních nedominovaných vektorů y^* budeme označovat Y_{PRN}.

Jak je vidět z poznámek, různí autoři bohužel používají různé označení a názvy pro uvedené pojmy. Někdy je pak tato nejednota poněkud matoucí. My se budeme držet našich definic a značení.

Poznámka: Množiny označujeme podle původních anglických názvů eficientních řešení. Například množina slabě eficientních řešení X_{WE} podle Weakly Efficient.

Příklad 2.3: [6] V příkladu 2.1 uvažujme řešení $x^* = (-1, 0)$. Ukážeme si, že toto eficientní řešení není vlastním. Musíme tedy ukázat, že pro každé $M > 0$ existuje index $k \in \{1, 2\}$ a $x \in X$ splňující $f_k(x) > f_k(x^*)$ tak, že

$$\frac{f_k(x) - f_k(x^*)}{f_j(x^*) - f_j(x)} > M$$

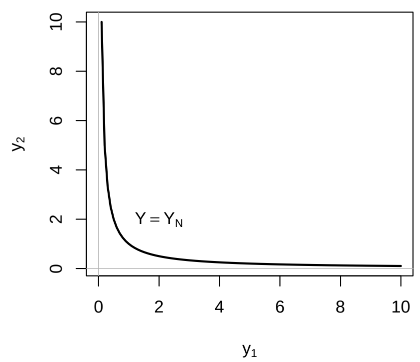
pro všechna $j \in \{1, 2\}$ taková, že $f_j(x^*) > f_j(x)$.

Vezměme $k = 1$ a zvolme x^ε takové, že $x_1^\varepsilon = -1+\varepsilon,\ 0 < \varepsilon < 1$ a $x_2^\varepsilon = -1+\sqrt{1-\varepsilon^2}$ $\Rightarrow x^\varepsilon$ je eficientní, protože $(x_1^\varepsilon+1)^2+(x_2^\varepsilon+1)^2 = 1$. Protože $x^\varepsilon \in X$, $x_1^\varepsilon > x_1^*$ a $x_2^\varepsilon < x_2^*$, máme $k = 1$, $j = 2$ a platí

$$\frac{f_k(x^\varepsilon) - f_k(x^*)}{f_j(x^*) - f_j(x^\varepsilon)} = \frac{-1+\varepsilon-(-1)}{-(-1+\sqrt{1-\varepsilon^2})} = \frac{\varepsilon}{1-\sqrt{1-\varepsilon^2}} \to \infty \text{ pro } \varepsilon \to 0.$$

Příklad 2.4: [6] Nechť $Y = \{y \in R^2 \mid y_1 > 0,\ y_2 = \frac{1}{y_1}\}$. Pak $Y_N = Y$, ale $Y_{PRN} = \emptyset$.

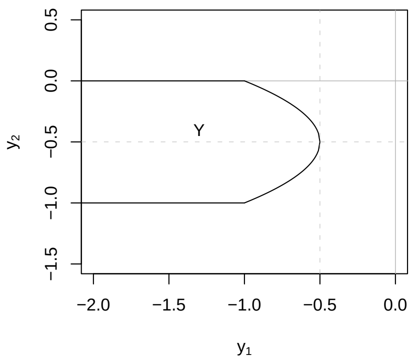
Vidíme to z toho, že vezmeme-li libovolné $y^* \in Y_N$, tak když se $y_1 < y_1^*$ blíží k 0, pak se y_2 limitně blíží k ∞. Podobně jde-li $y_2 < y_2^*$ k 0, pak se y_1 limitně blíží k ∞.



Obrázek 2.3: Ilustrace $Y = Y_N$, $Y_{PRN} = \emptyset$.

Příklad 2.5: [9] Nechť $X = \{x \in R^2 \mid x_1 + x_2 = 1,\ x_1 \geq 0,\ x_2 \geq 0\}$ a $f(x) = [-x_1^2 - x_2^2,\ -x_1]$. Potom

$$\begin{aligned} Y &= \{y \in R^2 \mid y_1 = -x_1^2 - x_2^2,\ y_2 = -x_1,\ x_1 + x_2 = 1,\ x_1 \geq 0,\ x_2 \geq 0\} \\ &= \{y \in R^2 \mid y_1 + \tfrac{1}{2} = -2(y_2 + \tfrac{1}{2})^2,\ -1 \leq y_2 \leq 0\}. \end{aligned}$$



Obrázek 2.4: Množina Y.

Z obrázku 2.4 vidíme, že jen body $y \in Y$, pro které platí $-1 \leq y_1 \leq -\frac{1}{2}$, $-\frac{1}{2} \leq y_2 \leq 0$, jsou obrazem množiny X_E a tedy

$$X_E = \{x \in X \mid 0 \leq x_1 \leq \frac{1}{2}\}.$$

Nyní budeme hledat množinu vlastních eficientních řešení X_{PRE}. Zvolíme libovolně $x^* \in X_E$ a hledáme nejdřív všechna $x \in X$, pro která je $f_1(x) > f_1(x^*)$.
Protože $x_1 + x_2 = 1$, je

$$-x_1^2 - x_2^2 = -2x_1^2 + 2x_1 - 1 = -2(x_1 - \frac{1}{2})^2 - \frac{1}{2},$$

a protože $0 \leq x_1^* \leq \frac{1}{2}$, je $f_1(x) > f_1(x^*)$ splněno právě pro ty body $x \in X$, pro něž je $x_1^* < x_1 < 1 - x_1^*$. Pro tyto body platí $-x_1 < -x_1^*$ a tedy $f_2(x) < f_2(x^*)$. Dále

$$\frac{-2x_1^2 + 2x_1 - 1 + 2(x_1^*)^2 - 2x_1^* + 1}{-x_1^* + x_1} = \frac{-2(x_1 - x_1^*)(x_1^* + x_1 - 1)}{-x_1^* + x_1} = -2(x_1^* + x_1 - 1) \leq 2,$$

odkud vidíme, že M musí být větší nebo rovno 2.

Následně hledáme všechna $x \in X$, pro která je $f_2(x) > f_2(x^*)$, tj. body z X, pro něž je $x_1 < x_1^*$. Pro tyto body platí

$$-2(x_1 - \frac{1}{2})^2 - \frac{1}{2} < -2(x_1^* - \frac{1}{2})^2 - \frac{1}{2}$$

a tedy

$$f_1(x) < f_1(x^*).$$

Dále

$$\frac{-x_1 + x_1^*}{-2(x_1^* - x_1)(x_1^* + x_1 - 1)} = \frac{1}{2(1 - x_1^* - x_1)} \leq \frac{1}{2(1 - 2x_1^*)}.$$

Protože $0 \leq x_1^* \leq \frac{1}{2}$, existuje pro každé $x^* \in X_E \setminus \{\frac{1}{2}, \frac{1}{2}\}$ konstanta

$$M = \min\{2, \frac{1}{2(1-2x_1^*)}\} > 0.$$

Pro $x^* = \{\frac{1}{2}, \frac{1}{2}\}$ žádné $M > 0$ neexistuje a tedy

$$X_{PRE} = X_E \setminus \{\frac{1}{2}, \frac{1}{2}\}.$$

Existence eficientních řešení implikuje existenci slabě eficientních řešení. Podobně z definic vidíme, že existence vlastních eficientních řešení implikuje existenci eficientních řešení. Opačně tato implikace opět platit nemusí. Dostáváme tedy tento vztah:

$$X_{PRE} \subset X_E \subset X_{WE}.$$

Samozřejmě také platí

$$Y_{PRN} \subset Y_N \subset Y_{WN}.$$

Často chceme vědět, zda určité řešení úlohy (2.1) je eficientní. Využijeme převedení této úlohy na následující jednokriteriální problém

$$\max \sum_{k=1}^{K} f_k(x) \tag{2.2}$$

$$\text{za podmínek } f_k(x) \geq f_k(x^*),\ k = 1, \ldots, K,\ x \in X.$$

Pro ověřování eficientnosti řešení platí následující věta.

Věta 2.4: [2] *Vektor $x^* \in R^n$ náleží do množiny X_E právě tehdy, když je optimálním řešením úlohy (2.2).*

Věta 2.5: [2] *Každé optimální řešení úlohy (2.2) náleží do množiny X_E.*

Rozdíl mezi těmito dvěma na první pohled podobnými větami je ten, že když první věta ukáže, že $x^* \notin X_E$, ale úloha (2.2) má nějaké jiné optimální řešení $x^{*'}$, pak podle druhé věty $x^{*'} \in X_E$.

Úloha (2.2) se nám hodí i k zjišťování, zda jsou množiny eficientních (X_E) a vlastních eficientních (X_{PRE}) řešení prázdné.

Věta 2.6: [2] *Předpokládejme, že $x^* \in X$ v úloze (2.2) a že tato úloha nemá konečnou maximální hodnotu. Potom $X_{PRE} = \emptyset$.*

Věta 2.7: [2] *Nechť každá funkce $f_k,\ k = 1, \ldots, K$ je konkávní na konvexní množině X. Předpokládejme, že $x^* \in X$ v úloze (2.2) a že tato úloha nemá konečnou maximální hodnotu. Potom pokud je množina*

$$Z^* = \{z^* \in R^K \mid z^* \leq f(x) \text{ pro nějaké } x \in X\}$$

uzavřená, platí $X_E = \emptyset$.

Věta 2.8: [2] *Nechť každá funkce $f_k,\ k = 1, \ldots, K$ je lineární a nechť X je polyedr (jedná se tedy o speciální úlohu popsanou v následující poznámce). Předpokládejme, že $x^* \in X$ v úloze (2.2) a že tato úloha nemá konečnou maximální hodnotu. Potom $X_E = \emptyset$.*

Poznámka: Uvažujme lineární případ úlohy (2.1) v podobě

$$\max f(x) = \max Cx \tag{2.3}$$

$$\text{za podmínek } Ax = b,\ x \in X$$

kde $C \in R^{K \times n}$, $A \in R^{m \times n}$ jsou dané matice, $b \in R^m$ je daný vektor.

Pro $K \geq 2$ tuto úlohu nazýváme úlohou lineárního vícekriteriálního programování. V tomto speciálním případě platí tvrzení

Věta 2.9: [12] *Je-li $x^* \in X$ eficientním řešením úlohy (2.3), potom je $x^* \in X$ také vlastním eficientním řešením úlohy (2.3).*

V lineárním případě tedy platí rovnost

$$X_E = X_{PRE}.$$

Poznámka: K zajímavému závěru došel při zkoumání vztahu mezi eficientními a vlastními eficientními řešeními úlohy (2.1) Arthur Geoffrion [8].

Pokud jsou funkce $f_k,\ k = 1, \ldots, K$ spojité a konkávní na uzavřené konvexní množině X, potom platí

$$Y_{PRN} \subseteq Y_N \subseteq \overline{Y_{PRN}}.$$

Za těchto podmínek je tedy funkční hodnota libovolného eficientního, ale ne vlastního eficientního řešení, limitou funkčních hodnot nějaké posloupnosti vlastních eficientních bodů. Základ této myšlence položili již Arrow a kol. [1].

Problém nalezení množin X_E, X_{WE} a X_{PRE} (tím pádem současně i množin Y_N, Y_{WN} a Y_{PRN}) úlohy (2.1) lze za určitých předpokladů převést na řešení úlohy parametrického programování, tentokrát však pouze s jedinou účelovou funkcí. Naši základní úlohu převedeme na tzv. parametrický skalární ekvivalent

$$\max_{x \in X} \sum_{k=1}^{K} \lambda_k f_k(x). \tag{2.4}$$

Parametry $\lambda_k,\ k = 1, \dots, K$ jsou váhy ohodnocující účelové funkce $f_k,\ k = 1, \dots, K$ z úlohy (2.1).

Je potřeba rozlišovat, zda jsou váhy nezáporné, nebo kladné. V případě nezáporných vah pracujeme s vektorem $\lambda = (\lambda_1, \dots, \lambda_K)$, $\lambda \in \Lambda_0$. Vždy alespoň jeden prvek je kladný, protože všechny váhy nulové nedávají příliš smysl. Kladné váhy označíme $\lambda = (\lambda_1, \dots, \lambda_K)$, $\lambda \in \Lambda$.
$\sum\limits_{k=1}^{K} \lambda_k = 1$ není nesplnitelné omezení, neboť libovolné váhy snadno znormujeme.
Poznámka: Tuto alternativu využíváme u tzv. přístupu váženého součtu optimalizovaných funkcí popisovaného v oddílu 2.3.2.

Řešení této úlohy pak mají vlastnosti shrnuté v následujících větách. Vidíme, že hlavní odlišnost vlastních eficientních řešení je v tom, že jim příslušné váhy λ jsou požadovány striktně kladné.

Věta 2.10: [2] *Nechť x^* je optimální řešení úlohy (2.4) pro $\lambda \in \Lambda_0$. Potom $x^* \in X_{WE}$.*

Poznámka: Pro lineární případ ($X \subset R^n$ je neprázdný konvexní polyedr a $f(x)$ jsou lineární funkce na X) platí rovnou $x^* \in X_E$.

Věta 2.11: [2] *Nechť každá funkce $f_k,\ k = 1, \dots, K$ je konkávní na konvexní množině X. Pak $x^* \in X_{WE}$ právě tehdy, když x^* je optimálním řešením úlohy (2.4) pro nějaké $\lambda \in \Lambda_0$.*

Věta 2.12: [2] *Nechť x^* je jediné optimální řešení úlohy (2.4) pro $\lambda \in \Lambda_0$. Potom $x^* \in X_E$.*

Věta 2.13: [2] *Nechť každá funkce $f_k,\ k = 1, \dots, K$ je konkávní na konvexní množině X. Jestliže $x^* \in X_E$, ale $x^* \notin X_{PRE}$, pak existuje vektor $\lambda \in \Lambda_0$ (s alespoň jednou složkou $\lambda_k = 0$) tak, že x^* je optimálním řešením úlohy (2.4).*

Věta 2.14: [8] *Nechť jsou dány váhy $\lambda \in \Lambda$. Je-li $x^* \in X$ optimálním řešením úlohy (2.4), potom $x^* \in X$ je vlastní eficientní řešení úlohy (2.1).*

Věta 2.15: [2] *Nechť každá funkce $f_k,\ k = 1, \dots, K$ je konkávní na konvexní množině X. Pak $x^* \in X_{PRE}$ právě tehdy, když x^* je optimálním řešením úlohy (2.4) pro nějaké $\lambda \in \Lambda$.*

K čemu jsou tato tvrzení dobrá? Jako jedno z hlavních použití se nabízí generování bodů pro množiny X_E, X_{WE} a X_{PRE} (a tedy Y_N, Y_{WN} a Y_{PRN}). Volbou různých přípustných konkrétních λ a následným řešením úlohy (2.4) získáme odpovídající námi hledané prvky množiny X_E, případně X_{WE}, X_{PRE}. Tímto způsobem, použijeme-li všechny přípustné váhy λ, získáme úplnou námi hledanou množinu eficientních řešení. Tento postup by však byl velmi numericky náročný. Lze použít zjednodušenou variantu – nagenerujeme rovnoměrným způsobem co nejvíce různých vah λ. Tím potom získáme odpovídající eficientní řešení, kterými proložíme křivku, na které by se měla nacházet všechna námi hledaná řešení.

Alternativní definice vlastních eficientních řešení

V části s definicemi jsme si uvedli definici vlastních eficientních řešení tak, jak ji zavedl v roce 1968 Arthur Geoffrion. Tato definice je nejrozšířenější a nejpoužívanější. Není však jediná. Existují i další (méně používané) definice vlastních eficientních řešení. Některé si nyní uvedeme a zároveň popíšeme vztahy mezi nimi.

Jedná se o definice, se kterými přišli matematici Borwein (1977), Benson (1979) a Kuhn s Tuckerem (1951). První dvě pouze zmíníme, Kuhn-Tuckerovu definici si ukážeme podrobněji. Tyto definice spolu s Geoffrionovou navzájem souvisí a jsou za určitých podmínek a předpokladů propojené (ekvivalentní).

Borweinova a Bensonova definice jsou popsány na stranách 52-56 v knize [6]. Bensonova definice je silnější než Borweinova.

My si nyní více přiblížíme Kuhn-Tuckerovu definici. Ta se hodí v případech, kdy musíme řešit vícekriteriální optimalizační úlohu s množinou přípustných řešení zadanou systémem algebraických omezení, tj.

$$X = \{x \in R^n \mid (g_1(x), \ldots, g_m(x)) \geq 0\}.$$

Předpokládáme, že účelové funkce $f_k,\ k = 1, \ldots, K$ a funkce $g_j,\ j = 1, \ldots, m$ vyjadřující omezení, jsou spojitě diferencovatelné. Pro jistotu připomeneme, že uvažujeme úlohu vícekriteriálního programování

$$\max\ f(x) \tag{2.5}$$

$$\text{za podm. } g(x) \geq 0,$$

kde $f : R^n \to R^K$ a $g : R^n \to R^m$.

Na tomto místě trochu odbočíme a ukážeme si způsob, jak lze řešit úlohu (2.5) pomocí tzv. sedlového bodu Lagrangeovy funkce

$$L(x, \mu) = \nu f(x) + \mu g(x),\ \nu \in R^K,\ \mu \in R^m.$$

Bod (x^*, μ^*) nazveme sedlovým bodem Lagrangeovy funkce $L(x, \mu)$, jestliže platí

$$L(x, \mu^*) \leq L(x^*, \mu^*) \leq L(x^*, \mu) \text{ pro všechna } x \in X \text{ a } \mu \geq 0.$$

To znamená, že sedlový bod maximalizuje Lagrangeovu funkci podle všech $x \in X$ a minimalizuje ji podle všech nezáporných multiplikátorů μ.

Dále řekneme, že vektor $x^* \in X$ je optimálním řešením úlohy (2.5), jestliže $g_j(x^*) \geq 0$, $j = 1, \ldots, m$ a neexistuje takové $x \in X$, splňující

$$
\begin{aligned}
f_i(x) &\geq f_i(x^*) && \forall i = 1, \ldots, K, \\
f_i(x) &> f_i(x^*) && \text{pro nějaká } i \in \{1, \ldots, K\}, \\
g_j(x) &\geq 0, && j = 1, \ldots, m.
\end{aligned}
$$

Vidíme, že se vlastně jedná o námi již zadefinované eficientní řešení. Vztahy mezi tímto optimálním řešením úlohy (2.5) a odpovídajícím sedlovým bodem zkoumá A. Takayama v [17]. Za určitých předpokladů je pro úlohu (2.5) první složka sedlového bodu (x^*, μ^*) ekvivalentní s optimálním řešením x^*. Tuto úlohu tedy lze řešit i pomocí Lagrangeovy funkce $L(x, \mu)$ a hledáním jejího sedlového bodu.

Nyní se vracíme ke Kuhn-Tuckerově definici vlastních eficientních řešení.

Definice 2.7: [6] Přípustné řešení $x^* \in X$ se nazývá vlastní eficientní řešení, jestliže je eficientní a když neexistuje takový vektor $u^* \in R^n$, splňující

$$
\begin{aligned}
\langle \nabla_x f_k(x^*), u^* \rangle &\geq 0 && \forall k = 1, \ldots, K, && (2.6) \\
\langle \nabla_x f_i(x^*), u^* \rangle &> 0 && \text{pro nějaká } i \in \{1, \ldots, K\}, && (2.7) \\
\langle \nabla_x g_j(x^*), u^* \rangle &\geq 0 && \forall j \in \mathcal{J}(x^*) = \{j = 1, \ldots, m \mid g_j(x^*) = 0\}. && (2.8)
\end{aligned}
$$

Množina $\mathcal{J}(x^*)$ je tzv. množina aktivních omezení. Opět zde implicitně předpokládáme eficientnost řešení (podobně jako u Geoffriona; na rozdíl od Borweina s Bensonem).

Co si představit pod zmiňovaným vektorem u^*? Existence vektoru u^* splňujícího (2.6) – (2.8) znamená, že posunem ve směru u^* od x^* žádná účelová funkce neklesá (2.6), nějaké z nich dokonce rostou (2.7) a vektor u^* míří do množiny přípustných řešení a ne ven z ní (2.8). Vektor u^* tedy udává přípustný směr růstu, zlepšení.

Abychom mohli ukázat ekvivalenci mezi Geoffrionovou a Kuhn-Tuckerovou definicí, musí být splněna nějaká omezení. Především se jedná o tzv. Kuhn-Tuckerovu podmínku regularity (*K-T constraint qualification*).

Definice 2.8: [6] Diferencovatelná úloha (2.5) splňuje v bodě $x^* \in X$ K-T podmínku regularity, když pro každé $u^* \in R^n$ splňující $\langle \nabla_x g_j(x^*), u^* \rangle \geq 0 \quad \forall j \in \mathcal{J}(x^*) = \{j = 1, \ldots, m \mid g_j(x^*) = 0\}$ existuje reálné $\bar{t} > 0$, funkce $\theta : [0, \bar{t}] \to R^n$ a $\alpha > 0$ tak, že $\theta(0) = x^*$, $\theta'(0) = \alpha u^*$ a $g(\theta(t)) \geq 0$ pro všechna $t \in [0, \bar{t}]$ (neboli $\theta(t) \in X$ pro všechna $t \in [0, \bar{t}]$).

Poznámka: V literatuře lze nalézt, že ve speciálním případu pro $X \subset R^n$ neprázdnou, konvexní a polyedrickou platí K-T podmínka regularity v každém bodě množiny X.

Splněním tohoto omezení docílíme toho, že množina

$$
U^* = \{u^* \in R^n \mid \langle \nabla_x f_k(x^*), u^* \rangle \geq 0 \; \forall k; \; \langle \nabla_x g_j(x^*), u^* \rangle \geq 0, \; j \in \mathcal{J}(x^*)\}
$$

vektorů u^* je prázdná.

Konečně jsme se tedy dostali k vyjádření vztahu mezi těmito definicemi.

Věta 2.16: [6] *Jestliže diferencovatelná úloha (2.5) splňuje K-T podmínku regularity v bodě x^* a x^* je vlastní eficientní podle Geoffriona, potom je také vlastní eficientní podle Kuhn-Tuckera.*

Na opačnou implikaci musíme klást další nároky. Není však nutná K-T podmínka regularity.

Věta 2.17: [6] *Nechť funkce $f_k, g_j : R^n \to R$ jsou konkávní a spojitě diferencovatelné $(\forall k, j)$. Předpokládejme, že x^* je vlastní eficientní podle Kuhn-Tuckera. Potom je x^* vlastní eficientní i podle Geoffriona.*

Uvažujme znovu parametrický skalární ekvivalent (2.4)

$$\max_{x \in X} \sum_{k=1}^{K} \lambda_k f_k(x).$$

Za určitých podmínek jsou shodná optimální řešení úlohy (2.4) a vlastní eficientní (podle Geoffriona) řešení úlohy (2.1). Z posledních dvou vět vidíme propojenost mezi vlastními eficientními řešeními podle Kuhn-Tuckera a Geoffriona. Jako důsledek se tedy nabízí

Věta 2.18: *Nechť funkce $f_k : R^n \to R$ jsou konkávní a spojitě diferencovatelné na konvexní množině X $(\forall k)$, funkce $g_j : R^n \to R$ jsou též konkávní a spojitě diferencovatelné $(\forall j)$. Dále nechť úloha (2.5) splňuje K-T podmínku regularity v bodě x^*. Potom $x^* \in X$ je vlastní eficientní řešení (podle Kuhn-Tuckera) úlohy (2.5) právě tehdy, když $x^* \in X$ je optimální řešení úlohy (2.4) pro nějaké $\lambda \in \Lambda$.*

Důkaz: Tvrzení plyne z věty 2.15 a věty 2.16.

Poznámka: I v případě vlastních eficientních řešení podle Kuhn-Tuckera podobně jako na straně 22 (a za oněch podmínek) platí, že funkční hodnota libovolného eficientního, ale ne vlastního eficientního řešení je limitou funkčních hodnot nějaké posloupnosti vlastních eficientních bodů.

2.3.2 Způsoby hledání eficientních řešení

Řešením úlohy vícekriteriálního programování se rozumí hledání celé množiny X_E, její podmnožiny, nebo jen jednoho kompromisního řešení. Pro uživatele je většinou zajímavé hledání "nejlepšího" eficientního řešení, aniž bychom znali celou množinu X_E. K tomu se využívají následující metody (většina z nich je uvedena v [5]):

A) Přístup váženého součtu optimalizovaných funkcí

Účelové funkce shlukneme pomocí vah do jediné funkce a převedeme tak vlastně vícekriteriální problém na úlohu jednokriteriální optimalizace. Důležitost vah (a tím i důležitost jednotlivých kriterií) si volíme sami podle daných omezení.

Nejprve si uvedeme větu, která nás navede na tento postup.

Věta 2.19: [5] *Nechť je $\emptyset \neq X$ kompaktní, $f_k(x),\ k = 1, \ldots, K$ spojité na X (tedy existuje optimální řešení). Funkce $h : R^k \to R$ (skalarizační funkce) je libovolná spojitá funkce neklesající v argumentech. Potom aspoň jedno optimální řešení úlohy*

$$\max_{x \in X} h(f_1(x), \ldots, f_K(x))$$

je eficientním řešením úlohy (2.1).

Je možné zvolit

$$h(f_1(x), \ldots, f_K(x)) = \sum_{k=1}^{K} \lambda_k f_k(x),$$

kde λ_k jsou váhy, které nám ohodnocují optimalizované funkce f_k. $\lambda = (\lambda_1, \ldots, \lambda_K)$, $\lambda \in \Lambda_0$.

Že lze tento přístup použít vidíme z vět 2.10 – 2.15 v předcházející části.

Poznámka: Tento postup je aplikován na Markowitzův model optimalizace portfolia, který je popsaný v oddílu 3.3 a v kapitole 4.

B) Přístup s ϵ - omezením

Zvolíme kteroukoliv funkci z uvažovaných optimalizovaných funkcí $f_1, \ldots, f_K$. Možnosti výběru jsou různé. Lze ji vybrat podle různých kriterií, nebo i zcela náhodně. Bez újmy na obecnosti můžeme předpokládat, že vybraná funkce je f_1. Pro zbylé funkce zadáme dolní mez $\epsilon \in R^{K-1}$, $\epsilon = (\epsilon_2, \ldots, \epsilon_K)$ a požadujeme splnění podmínky $f_k(x) \geq \epsilon_k,\ k = 2, \ldots, K$. Jde tedy o to, že maximalizujeme zvolenou funkci f_1 za podmínky, že ostatní funkce dosahují minimálně námi zadanou dolní mez. Řešíme potom optimalizační úlohu

$$\max f_1(x) \quad \text{na } X_\epsilon = X \cap \bigcap_{k=2}^{K} \{x \mid f_k(x) \geq \epsilon_k\}. \tag{2.9}$$

Pokud úloha (2.9) platí pro taková $\epsilon_2, \ldots, \epsilon_K : X_\epsilon \neq \emptyset$, potom platí následující dvě tvrzení:

Věta 2.20: [5] *Jestliže $x^* \in X$ je jediné optimální řešení úlohy (2.9), pak $x^* \in X$ je eficientní řešení úlohy (2.1).*

Věta 2.21: [5] *Jestliže $x^* \in X$ je eficientní řešení úlohy (2.1), pak existuje $\epsilon \in R^{K-1}$ takové, že $x^* \in X$ je optimální řešení úlohy (2.9).*

C) Cílové programování

Tento způsob použijeme v situacích, kdy známe od uživatele hodnoty cílových funkcí (hodnoty jednotlivých kriterií), jichž by si přál dosáhnout. K výsledku se lze dopracovat více způsoby. Záleží na tom, zda uživatel uvede preference dosažení jednotlivých cílových hodnot, jak budeme definovat funkci určující odchylky od cílů (tu budeme samozřejmě minimalizovat) atd.

Matematicky tento způsob formulujeme následovně. Označme si $f^* = (f_1^*, \ldots, f_K^*)$ vektor funkčních hodnot, ke kterým chceme dojít. Chceme tedy najít v množině přípustných řešení X takové $\tilde{x}$, aby jednotlivé účelové funkce splňovaly

$$f_1(\tilde{x}) = f_1^*, \ldots, f_K(\tilde{x}) = f_K^*, \text{ tj. } f(\tilde{x}) = f^*.$$

Přejeme si minimalizovat odchylku (vzdálenost) mezi $f(x)$ a f^*

$$\min_{x \in X} d(f^*, f(x)).$$

Vzdálenost vyjádříme pomocí normy

$$d(f^*, f(x)) = \|f^* - f(x)\|.$$

Podle zvolené normy existují různé modely cílového programování:

1. $\min\limits_{x \in X} \{\|f(x) - f^*\|_p = [\sum\limits_{k=1}^{K} |f_k(x) - f_k^*|^p]^{\frac{1}{p}}\}$

2. $\min\limits_{x \in X} \{\|f(x) - f^*\|_1 = \sum\limits_{k=1}^{K} |f_k(x) - f_k^*|\}$

3. $\min\limits_{x \in X} \{\|f(x) - f^*\|_2 = [\sum\limits_{k=1}^{K} |f_k(x) - f_k^*|^2]^{\frac{1}{2}}\}$

3*. $\lambda_k,\ k = 1 \ldots K$ jsou váhové koeficienty
$\min\limits_{x \in X} \{\|f(x) - f^*\|_2 = [\sum\limits_{k=1}^{K} \lambda_k |f_k(x) - f_k^*|^2]^{\frac{1}{2}}\}$

4. $\min\limits_{x \in X} \{\|f(x) - f^*\|_\infty = \max\limits_{1 \le k \le K} |f_k(x) - f_k^*|\}$

Pro lepší představu uvedeme příklad, kdy se chceme funkcemi $f_k(x),\ k = 1, \ldots, K$ co nejvíce přiblížit ideálním (cílovým) hodnotám $f_1^*, \ldots, f_K^*$. Odchylky $f_k(x)$ od f_k^* označíme d_k^+ (když $f_k(x) > f_k^*$, tj. pokud je odchylka kladná), nebo d_k^- ($f_k(x) < f_k^*$, tj. odchylka je záporná). Pak tedy řešíme úlohu

$$\min D^*$$

za podmínek $f_k(x) + d_k^- + d_k^+ = f_k^*,\ d_k^+ \ge 0,\ d_k^- \le 0,\ d_k^+ \cdot d_k^- = 0,\ k = 1, \ldots, K.$

D^* je funkce, kterou si můžeme sestavit libovolně pomocí odchylek d_k^+ a d_k^-. Záleží na nás, jaké máme požadavky a preference.

Lze minimalizovat pouze $d^+ = \sum_{k=1}^{K} d_k^+$. Tím se bude výsledné řešení co nejvíce shora přibližovat cílové hodnotě a může být libovolně nižší. Pokud budeme naopak minimalizovat samotné $d^- = \sum_{k=1}^{K} d_k^-$, výsledné řešení se bude logicky co nejvíce zdola blížit k cílové hodnotě a může být samozřejmě libovolně vyšší. Kdybychom minimalizovali současně $d^+ + d^-$, docílíme toho, že výsledek se bude blížit k cílovému řešení jak shora, tak zdola.

Uživatel může mít i požadavky ohledně preference jednotlivých funkcí f_k. To lze vyjádřit přiřazením vah k příslušným odchylkám d_k podobně jako v A).

D) Jiné přístupy

Smíšený postup

Tento přístup je kombinací váženého součtu optimalizovaných funkcí a přístupu s ϵ - omezením. Některé funkce se přesunou do podmínek a omezení nové úlohy, ostatním se přiřadí váhy.

Lexikografická metoda

Funkce jsou seřazeny podle důležitosti a optimalizuje se nejprve jen podle nejdůležitější z nich. Pokud zůstane více podobných řešení, pokračuje se (s touto omezenou množinou přípustných řešení) druhou nejdůležitější funkcí. Takto postupujeme, dokud nezbyde jediné řešení.

Shlukování funkcí do jedné

Shlukování funkcí se podobá váženému součtu optimalizovaných funkcí. Také se nějakým mechanismem (pomocí koeficientů) převede vícekriteriální problém na jednokriteriální úlohu. Jak je patrné z názvu metody, funkce se do jediné "shlukují".

Příklad:

1. $h(f_1, \ldots, f_K) = \sum_{k=1}^{K} \dot{\alpha}_k [f_k(x)]^{\dot{\beta}_k}, \quad \dot{\alpha}_k \geq 0,\ \dot{\beta}_k > 0$

2. $h(f_1, \ldots, f_K) = -\sum_{k=1}^{K} \ddot{\alpha}_k \exp[-f_k(x)], \quad \ddot{\alpha}_k \geq 0$

Simplexová metoda [6]

Tuto metodu lze použít pouze v případě lineárních účelových funkcí. Účelové funkce shlukneme do jedné pomocí parametrů a pokračujeme jako při jednokriteriální simplexové metodě – nalezneme výchozí řešení, které (pokud není optimální) postupně zlepšujeme.

Kapitola 3

Vícekriteriální stochastická optimalizace

Doposud jsme se zabývali pouze deterministickými vícekriteriálními optimalizačními úlohami. Všechny koeficienty a hodnoty funkcí vyskytující se v zadání byly předem známé. Pomocí nich se daly úlohy vyřešit.

Problém ovšem nastává, když koeficienty (parametry) neznáme předem s určitostí, ale známe pouze jejich rozdělení. Neboli když funkce závisí na náhodném parametru, nějaký z koeficientů má náhodný charakter. A po nás se tedy požaduje učinit rozhodnutí dříve, než máme jejich hodnoty k dispozici. V tom případě se jedná o úlohu stochastického programování. Základní informace a přehled stochastického jednokriteriálního programování je k nalezení například v [4].

Jestliže řešení úlohy má být určeno bez znalosti realizace náhodných veličin, musíme tento náš problém převést na úlohu deterministické optimalizace. Tento přístup většinou zaručuje přijatelné hodnoty pro všechna kriteria.

3.1 Úvod

Vícekriteriální optimalizační úlohu s náhodným elementem obecně vyjádříme jako

$$\max\left[f_1(x,\omega),\dots,f_K(x,\omega)\right] \tag{3.1}$$

$$\text{za podmínek } g(x,\omega)\geq 0,\ x\in X,$$

kde $X\subset R^n$ je neprázdná množina, ω je s-rozměrný náhodný vektor z daného pravděpodobnostního prostoru $(\Omega,\mathcal{A},P)$, $f_k: R^n\times\Omega\to R,\ k=1,\dots,K$ a $g: R^n\times\Omega\to R^m$ jsou dané reálné funkce.

Příklad 3.1: Jako jednoduchý jednokriteriální příklad pro ilustraci a představu o stochastické optimalizaci nám poslouží velmi známá "úloha prodavače novin".

Představme si prodavače novin, který stojí například na nějakém nádraží. Každý den stojí na stejném místě a prodává noviny. Pro jednoduchost předpokládejme, že nabízí pouze jeden druh periodika. Mohli bychom jeho nabídku rozšířit na více druhů, ale tím by se úloha zesložitila (lidé by si totiž mohli v případě vyprodání jednoho druhu koupit "náhradní" druh). Samozřejmě dopředu nemůže vědět, kolik zájemců o noviny se u něj za den zastaví. Toto je v našem příkladě onen náhodný prvek. Počet potenciálních kupujících tvoří kompaktní nosič $[b_*, b^*]$. Dá se totiž předpokládat, že během dne přijde určitě alespoň b_*, maximálně však b^* zájemců.

Prodavač odebere x výtisků deníku v ceně c^N za kus. Sám je však prodává za cenu c^P za kus. Je logické, že $c^P > c^N > 0$. Pokud mu na konci dne nějaké noviny zbydou (neboli poptávka je menší než počet nabízených novin), mají zbylé výtisky nulovou cenu (tedy 0 za kus).

Otázka pro prodavače zní, kolik má objednat novin, aby v průměru získal maximální očekávaný výnos? Pokud má novin moc, tak mu jich hodně zbyde a není to příliš výhodné. Pokud by však novin objednal málo a rychle je vyprodá, sám se připravuje o další zisk a navíc se může stát, že zákazník, na kterého se již nedostalo, už třeba příště nepřijde.

Zápis:
x ... počet výtisků odebraných prodavačem,
c^N ... nákupní cena jednoho výtisku,
c^P ... prodejní cena jednoho výtisku,
ω ... náhodná poptávka; rozdělení poptávky ω je P,
$X = [b_*, b^*]$ ... předpokládaný nosič; $0 < b_* < b^*$.

Podle počtu objednaných novin a poptávky má prodavač výnos
$-c^N x + c^P \omega$ ... pokud $\omega \leq x$,
$-c^N x + c^P x$ ... pokud $\omega > x$.
Funkci, kterou chceme maximalizovat, tak zapíšeme
$(c^P - c^N)x - c^P(x-\omega)^+$.

Prodavač tento problém řeší každý den, proto je rozumné použít střední hodnotu. Předpokládáme však, že má dostatečné zásoby peněz pro případ, kdyby se mu ze začátku příliš nedařilo. Řešíme tedy úlohu

$$\max_{x \in X} [(c^P - c^N)x - c^P\, E_P(x-\omega)^+].$$

Předpokládejme, že rozdělení P je absolutně spojité s hustotou f_P a distribuční funkcí F. Pak střední hodnotu našeho problému vypočítáme takto:

$$E_P(x-\omega)^+ = \int_{b_*}^{b^*} (x-\hat{z})^+ f_P(\hat{z})\,\mathrm{d}\hat{z} = \begin{cases} 0, & x < b_* \\ x - E_P\,\omega, & x > b^* \\ \bigotimes, & x \in [b_*, b^*] \end{cases}$$

$$\bigotimes = \int_{b_*}^{x} (x-\hat{z}) f_P(\hat{z})\,\mathrm{d}\hat{z} = xF(x) - \int_{b_*}^{x} \hat{z} f_P(\hat{z})\,\mathrm{d}\hat{z} = xF(x) - [\hat{z}F(\hat{z})]_{b_*}^{x} + \int_{b_*}^{x} F(\hat{z})\,\mathrm{d}\hat{z} =$$
$$= xF(x) - xF(x) + b_*F(b_*) + \int_{b_*}^{x} F(\hat{z})\,\mathrm{d}\hat{z} = b_*F(b_*) + \int_{b_*}^{x} F(\hat{z})\,\mathrm{d}\hat{z}$$

Z toho derivace podle x je $F(x)$, tudíž

$$c^P - c^N - c^P F(x) = 0,$$

$$F(x) = \frac{c^P - c^N}{c^P} = 1 - \underbrace{\frac{c^N}{c^P}}_{\alpha} \rightarrow \underline{x = F^{-1}(1-\alpha)}.$$

Pro názornost si můžeme předvést konkrétní rozdělení, například rovnoměrné rozdělení na $[b_*, b^*]$. V tomto případě

$$\bigotimes = \frac{1}{b^* - b_*} \int\limits_{b_*}^{x} (x - \hat{z})\,\mathrm{d}\hat{z} = \frac{(x - b_*)^2}{2(b^* - b_*)}.$$

3.2 Přístupy k řešení vícekriteriálních stochastických úloh

Jak již bylo řečeno, stochastické úlohy se převádí na deterministické. Tato transformace se uskutečňuje pomocí různých statistických charakteristik náhodných veličin, jako například pomocí střední hodnoty. Podle toho, které charakteristiky využijeme, můžeme získat různá řešení pro jednu a tu samou původní stochastickou úlohu.

3.2.1 Náhoda pouze v účelové funkci

Uvažujme vícekriteriální optimalizační úlohu s náhodným elementem

$$\max_{x \in X} [f_1(x,\omega), \ldots, f_K(x,\omega)], \tag{3.2}$$

kde $X \subset R^n$ je deterministická množina přípustných řešení. Na náhodě tedy nezávisí. Hodnota každé účelové funkce zde závisí na s-rozměrném vektoru náhodných parametrů ω se známým rozdělením P. Tuto hodnotu tedy známe až po realizaci náhodného vektoru ω. Pro dané $x \in X$ a různé realizace ω se hodnoty jednotlivých $f_k(x,\omega),\ k = 1,\ldots,K$ obecně liší. Proto je k výběru rozhodnutí $x \in X$ vhodné použít transformovanou funkci $\psi(x)$, která může být jednokriteriální i vícekriteriální, ale na náhodě může záviset pouze prostřednictvím příslušné pravděpodobnostní míry.

Jako jednoduchý příklad každé účelové funkce $f_k(x,\omega)$ vezmeme lineární funkci $c_k^T x$, kde náhodné jsou koeficienty n-rozměrného vektoru c_k ($c_k := c_k(\omega)$, kde $c_k(\omega)$ je měřitelná) s rozdělením označeným P. U jednotlivých případů si uvedeme výsledný tvar účelové funkce při obecném rozdělení P i při konkrétním rozdělení $c_k \sim N_n(\mu_k, \Sigma_k)$, kde μ_k je n-rozměrný vektor a Σ_k je matice typu $n \times n$. Vše platí pro $k = 1,\ldots,K$.

Podle různých voleb funkce $\psi(x)$ rozlišujeme následující způsoby transformace stochastické úlohy (3.2) na deterministické úlohy. Převádíme samozřejmě za předpokladu, že inkriminovanou transformaci lze uskutečnit (požadované statistické charakteristiky náhodných veličin existují).

E-kriterium (Bayesovo kriterium)

První koncepcí je využít střední hodnoty náhodných veličin. Jde o to, že náhody v původní stochastické úloze nahradíme jejich střední hodnotou. Původní $f_k(x,\omega)$ se tedy nahradí $E_P\, f_k(x,\omega),\ k = 1,\ldots,K$.

Poznámka: Střední hodnotu je někdy nutné počítat u upravené funkce $\psi(x) = E_P\, z\{f(x,\omega)\}$, kde $z : R \to R$ je vhodná ztrátová funkce, jejíž střední hodnota existuje pro všechna $x \in X$.

Příklad:

$$\max_{x\in X} \psi(x) \sim \max_{x\in X} E_P\,[f_1(x,\omega),\ldots,f_K(x,\omega)] \sim \max_{x\in X} E_P[c_1^T x,\ldots,c_K^T x]$$
$$\sim \max_{x\in X} [\mu_1^T x,\ldots,\mu_K^T x]$$

P-kriterium (Royovo kriterium, "safety first")

Zde máme zadané hodnoty $u_1,\ldots,u_K$ a maximalizujeme pravděpodobnost toho, že příslušné účelové funkce budou dosahovat nejméně takových hodnot. Původní $f_k(x,\omega)$ se tedy nahradí $P\{f_k(x,\omega) \geq u_k\},\ k = 1,\ldots,K$.

Příklad:

$$\max_{x\in X} \psi(x) \sim \max_{x\in X} [P\{f_1(x,\omega) \geq u_1\},\ldots,P\{f_K(x,\omega) \geq u_K\}]$$
$$\sim \max_{x\in X} [P\{c_1^T x \geq u_1\},\ldots,P\{c_K^T x \geq u_K\}]$$
$$\sim \max_{x\in X} [\tfrac{\mu_1^T x - u_1}{\sqrt{x^T\Sigma_1 x}},\ldots,\tfrac{\mu_K^T x - u_K}{\sqrt{x^T\Sigma_K x}}]$$

Pro různá $x, x' \in R^n,\ x \neq x'$ a libovolná $k = 1,\ldots,K$ platí

$$P\{f_k(x,\omega) \geq u_k\} > P\{f_k(x',\omega) \geq u_k\} \Longleftrightarrow \frac{\mu_k^T x - u_k}{\sqrt{x^T\Sigma_k x}} > \frac{\mu_k^T x' - u_k}{\sqrt{x'^T\Sigma_k x'}}.$$

Pak tedy lze transformovat stochastické úlohy na deterministické uvedeným způsobem.

U tohoto kriteria si pro názornost uvedeme, jak dospějeme od $\max_{x\in X} P\{c_k^T x \geq u_k\}$ k $\max_{x\in X} \frac{\mu_k^T x - u_k}{\sqrt{x^T\Sigma_k x}}$ při mnohorozměrném rozdělení $c_k \sim N_n(\mu_k,\Sigma_k)$.

V tomto případě má $c_k^T x$ jednorozměrné normální rozdělení $N(\mu_k^T x, \sigma_k^2(x))$, což označíme jako $\xi_k \sim N(\mu_k^*, \sigma_k^2)$.

$$P\{\xi_k \geq u_k\} = 1 - P\{\xi_k \leq u_k\} = 1 - P\{\underbrace{\frac{\xi_k - \mu_k^*}{\sigma_k}}_{\sim N(0,1)} \leq \frac{u_k - \mu_k^*}{\sigma_k}\} = 1 - \Phi(\frac{u_k - \mu_k^*}{\sigma_k}),$$

kde Φ je známé označení distribuční funkce u normovaného normálního rozdělení $N(0,1)$.

$$\max_{x \in X} [1 - \Phi(\frac{u_k - \mu_k^T x}{\sqrt{x^T \Sigma_k x}})] \sim \min_{x \in X} \Phi(\frac{u_k - \mu_k^T x}{\sqrt{x^T \Sigma_k x}}) \sim \min_{x \in X} \frac{u_k - \mu_k^T x}{\sqrt{x^T \Sigma_k x}} \sim \max_{x \in X} \frac{\mu_k^T x - u_k}{\sqrt{x^T \Sigma_k x}}$$

Poznámka: Při uvedeném postupu je potřeba klást požadavek na nenulovost $x \in X$.

Kvantilové kriterium (Kataokovo pravděpodobnostní kriterium)

Máme zadané pravděpodobnosti $\beta_1, \ldots, \beta_K$ a hledáme maximální hodnoty $u_1, \ldots, u_K$ takové, aby daná účelová funkce $f_k(x, \omega)$ dosahovala alespoň příslušné hodnoty u_k s danou pravděpodobností $\beta_k \in (0, 1]$, $k = 1, \ldots, K$.

Příklad:

$$\begin{aligned}
\max \psi(x) \sim\ & \max\ (u_1, \ldots, u_K) \\
& \text{za podmínek } P\{f_k(x, \omega) \geq u_k\} \geq \beta_k, \quad k = 1, \ldots, K \\
& \qquad\qquad\quad x \in X \\[1em]
\sim\ & \max\ (u_1, \ldots, u_K) \\
& \text{za podmínek } P\{c_k^T x \geq u_k\} \geq \beta_k, \quad k = 1, \ldots, K \\
& \qquad\qquad\quad x \in X \\[1em]
\sim\ & \max\ (u_1, \ldots, u_K) \\
& \text{za podmínek } u_k \leq \mu_k^T x + \Phi^{-1}(1 - \beta_k)\sqrt{x^T \Sigma_k x}, \quad k = 1, \ldots, K \\
& \qquad\qquad\quad x \in X
\end{aligned}$$

kde Φ^{-1} je známé označení kvantilu $N(0, 1)$. Hodnota každého $\beta_k \geq 0{,}5$ (tedy $\Phi^{-1}(1 - \beta_k) \leq 0$) vede na úlohu konvexního programování, neboť $\mu_k^T x$ je lineární a $\sqrt{x^T \Sigma_k x}$, $k = 1, \ldots, K$ jsou konvexní [15].

Poznámka: Je třeba podotknout, že tímto dojde k přesunutí náhodných parametrů do omezení.

V-kriterium (rizikové kriterium)

Dalším způsobem je použití rozptylů náhodných veličin, které chceme samozřejmě minimalizovat. Původní $f_k(x, \omega)$ se tedy nahradí $var(f_k(x, \omega))$, $k = 1, \ldots, K$.

Příklad:

$$\min_{x\in X} \psi(x) \sim \min_{x\in X} [var(f_1(x,\omega)),\ldots,var(f_K(x,\omega))]$$
$$\sim \min_{x\in X} [var(c_1^T x),\ldots,var(c_K^T x)]$$
$$\sim \min_{x\in X} [x^T\Sigma_1 x,\ldots,x^T\Sigma_K x]$$

Převedení této koncepce do tvaru blízkého kvantilovému kriteriu v případě, že pracujeme s lineárními funkcemi, je uvedeno v [16]. Proti riziku se můžeme bránit i tak, že rozptyl přesuneme do podmínek.

Uvažujme, že funkce $f_k(x,\omega)$ má tvar $c_k^T x$, kde figurují již představené náhodné koeficienty c_k, $k=1,\ldots,K$. Úloha minimalizace

$$[var(c_1^T x),\ldots,var(c_K^T x)] = [E_P\,(c_1^T x - \bar{c_1}^T x)^2,\ldots,E_P\,(c_K^T x - \bar{c_K}^T x)^2]$$

se přetransformuje do tvaru

$$\min\,(u_1,\ldots,u_K)$$

za podmínek

$$P\{(c_k^T x - \bar{c_k}^T x)^2 \leq u_k\} = \delta_k, \quad k=1,\ldots,K,$$
$$x \in X,$$

kde $\bar{c_k}$ je obecně střední hodnota vektoru c_k, u_k je horní přípustná mez pro $(c_k^T x - \bar{c_k}^T x)^2$ a $\delta_k \in (0,1]$ je daná pravděpodobnost, se kterou nesmí být horní mez u_k překročena. Vždy $k=1,\ldots,K$.

Jde tedy vlastně o kvantilové kriterium, kde funkce $f_k(x,\omega)$ mají tvar $(c_k^T x - \bar{c_k}^T x)^2$. Opět tedy přesouváme náhodné parametry do omezení. Jediným rozdílem je, že zde máme u pravděpodobnosti rovnost místo nerovnosti použité u kvantilového kriteria.

Jestliže c_k, $k=1,\ldots,K$ mají normální rozdělení, lze model převést a řešit pomocí χ^2-rozdělení [16].

Smíšené kriterium (E kriterium + riziko)

V tomto přístupu využíváme současně střední hodnoty a směrodatné odchylky všech stochastických účelových funkcí. V přetransformované deterministické úloze tedy pracujeme s dvojnásobkem účelových funkcí. Střední hodnoty maximalizujeme, směrodatné odchylky bychom pochopitelně rádi minimalizovali. Původní $f_k(x,\omega)$ nahradíme dvěma hodnotami $E_P\,f_k(x,\omega)$ a $-\sqrt{var(f_k(x,\omega))}$, $k=1,\ldots,K$.

Poznámka: Omezíme se pouze na případ lineárních funkcí, který je nejvýznamnější.

Příklad:

$$\max_{x\in X} \psi(x) \sim \max_{x\in X} [E_P(c_1^T x),\ldots,E_P(c_K^T x), -\sqrt{var(c_1^T x)},\ldots,-\sqrt{var(c_K^T x)}]$$
$$\sim \max_{x\in X} [\mu_1^T x,\ldots,\mu_K^T x, -\sqrt{x^T\Sigma_1 x},\ldots,-\sqrt{x^T\Sigma_K x}]$$

Jak vidíme, funkci $\psi(x)$ lze volit různými způsoby. Záleží především na povaze řešené úlohy. Pokud máme dostatečné rezervy a úlohu řešíme opakovaně, lze použít E- kriterium. Pokud jsou ovšem naše rezervy minimální, je tento způsob řešení nevhodný a použijeme například P-kriterium.

Pomocí těchto pěti přístupů (E-kriterium, P-kriterium, kvantilové kriterium, V- kriterium a smíšené kriterium) dojde k převedení stochastické úlohy (3.2) na odpovídající deterministické. U těch již umíme zjistit množiny eficientních řešení. Množinu eficientních řešení deterministické úlohy získané pomocí E-kriteria (resp. P-kriteria, kvantilového kriteria, V-kriteria a smíšeného kriteria) označíme $X_{\bar{E}}$ (resp. $X_{MR(u)}$, $X_{K(\beta)}$, X_{σ^2} a $X_{\bar{E}\sigma}$). Nyní si uvedeme nějaké vlastnosti těchto množin a popíšeme si vztahy mezi nimi. Budeme předpokládat, že deterministická množina přípustných řešení X je neprázdná, kompaktní a konvexní.

Je pochopitelné, že množina eficientních řešení $X_{MR(u)}$ závisí na hodnotách vektoru u. Pro jiný vektor $u' \neq u$ tak můžeme dostat jinou množinu $X_{MR(u')} \neq X_{MR(u)}$.

Jak získat různými způsoby množinu $X_{MR(u)}$ se zabývá Stancu-Minasian [16]. Omezuje se na lineární funkce a jejich zevšeobecnění (vynásobování, podíly lineárních funkcí).

Podobně i množina eficientních řešení $X_{K(\beta)}$ závisí na hodnotách vektoru pravděpodobností β. Pro jiný vektor $\beta' \neq \beta$ tak můžeme dostat jinou množinu $X_{K(\beta')} \neq X_{K(\beta)}$.

Jak získat různými způsoby množinu $X_{K(\beta)}$ se taktéž můžeme dočíst v [16]. Opět se to týká lineárních funkcí a jejich zevšeobecnění.

Nyní si uvedeme vztahy mezi množinami $X_{\bar{E}}$, X_{σ^2}, $X_{\bar{E}\sigma}$, $X_{MR(u)}$ a $X_{K(\beta)}$, které jsou představeny a dokázány v Caballero a kol. [3]. Autoři jdou více do hloubky a ukazují i vztahy mezi slabě eficientními a vlastními eficientními verzemi množin eficientních řešení.

Vztah mezi $X_{\bar{E}}$, X_{σ^2} a $X_{\bar{E}\sigma}$

Věta 3.1: *Každé řešení, které náleží do množiny $X_{\bar{E}}$ a zároveň i do množiny X_{σ^2}, tak je potom také prvkem množiny $X_{\bar{E}\sigma}$.*

$$X_{\bar{E}} \cap X_{\sigma^2} \subset X_{\bar{E}\sigma}$$

Jako názorná ukázka tohoto vztahu nám může posloužit následující příklad.

Příklad 3.2: [3] Představme si úlohu

$$\max_{x \in X} [c_1^T x,\; c_2^T x],$$

kde

$$X = \{x \in R^2 \mid x_1 + x_2 \geq 1,\ x_1 + 3x_2 \leq 10,\ -2 \leq -x_1 + x_2 \leq 2,\ x_1 \geq 0,\ x_2 \geq 0\},$$

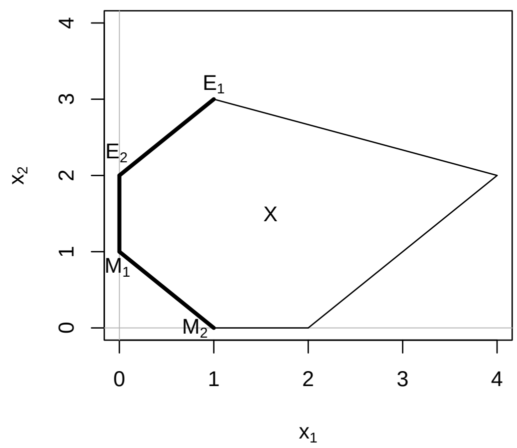
$$c_i = c_i^1 + t_i c_i^2, \quad i = 1, 2.$$

Tyto body nabývají hodnot

$$c_1^1 = (7, 12)^T,\ c_1^2 = (-6, -5)^T,\ c_2^1 = (-3, 5)^T,\ c_2^2 = (-4, -8)^T.$$

V úloze figurují náhodné veličiny t_1 a t_2, kde t_1 má normální rozdělení $t_1 \sim N(1, 4)$ a t_2 má exponenciální rozdělení $t_2 \sim Ex(2)$.
Z těchto údajů přetransformujeme stochastickou úlohu na deterministickou pomocí smíšeného kriteria.

$$\max_{x \in X} [x_1 + 7x_2,\ -5x_1 + x_2,\ -12x_1 - 10x_2,\ -2x_1 - 4x_2].$$



Obrázek 3.1: Grafické znázornění řešení.

Body $E_1 = (1, 3)$ a $E_2 = (0, 2)$, resp. $M_1 = (0, 1)$ a $M_2 = (1, 0)$ jsou optimální řešení deterministické úlohy získané pomocí E-kriteria, resp. V-kriteria. Z toho plyne, že úsečka E_1E_2 je množina eficientních bodů $X_{\bar{E}}$ a část M_1M_2 je množina eficientních bodů X_{σ^2}. Eficientní řešení z množiny $X_{E\sigma}$ se nachází na úsečkách M_1M_2, M_1E_2, E_2E_1. Tedy $X_{E\sigma}$ zahrnuje jak $X_{\bar{E}}$, tak i X_{σ^2}.

Vztah mezi $X_{\bar{E}\sigma}$ a $X_{K(\beta)}$

Na úvod poznamenejme, že $X_{MR(u)}$ a $X_{K(\beta)}$ jsou na reciproční bázi. Budeme se zabývat vztahy mezi množinami eficientních řešení $X_{\bar{E}\sigma}$ a $X_{K(\beta)}$ při lineárních účelových funkcích původní úlohy. Ukážeme si dva případy, pro které jsou uvedené vztahy odvozeny.

V prvním případě předpokládáme, že $f_k(x,\omega) = c_k^T x,\ \ k = 1,\dots,K$, kde c_k je náhodný vektor s normálním rozdělením, tentokrát se střední hodnotou $\bar{c}_k$ a pozitivně definitní varianční maticí V_k. Dále předpokládáme, že $0 \notin X$.
Za těchto předpokladů označíme střední hodnotu funkce $f_k(x,\omega)$

$$\bar{f}_k(x) = \bar{c}_k^T x$$

a její směrodatnou odchylku

$$\sigma_k(x) = \sqrt{x^T V_k x}.$$

Distribuční funkce pro mez u_k je potom

$$P\{c_k^T x \geq u_k\} = 1 - \Phi(\frac{u_k - \bar{c}_k^T x}{\sqrt{x^T V_k x}}),$$

kde Φ označuje distribuční funkci standardizovaného normálního rozdělení $N(0,1)$.
Pravděpodobnostní omezení

$$P\{c_k^T x \geq u_k\} \geq \beta_k, \quad \beta_k \in (0,1]$$

tak odpovídá nerovnosti

$$\bar{c}_k^T x + \Phi^{-1}(1-\beta_k)\sqrt{x^T V_k x} \geq u_k.$$

Tuto nerovnost přepíšeme do tvaru

$$\bar{f}_k(x) + \alpha_k \sigma_k(x) \geq u_k, \quad \alpha_k = \Phi^{-1}(1-\beta_k).$$

Když $\alpha_k = \Phi^{-1}(1-\beta_k)$, tak potom $\alpha_k < (\geq)\, 0$, pokud $\beta_k > (\leq)\, 0{,}5$.

V druhém případě předpokládejme, že $f_k(x,\omega) = c_k^T x,\ \ k = 1,\dots,K$, kde c_k je opět náhodný vektor, který nyní lineárně závisí na náhodné veličině t_k ($c_k = c_k^1 + t_k c_k^2$). Střední hodnota veličiny t_k bude $\bar{t}_k$, její směrodatná odchylka υ_k, $\upsilon_k < \infty$ a F_k bude její rostoucí distribuční funkce. Dále předpokládáme, že pro každé $x \in X$ platí ${c_k^2}^T x > 0$.
Za těchto předpokladů označíme střední hodnotu funkce $f_k(x,\omega)$

$$\bar{f}_k(x) = {c_k^1}^T x + \bar{t}_k {c_k^2}^T x$$

a její směrodatnou odchylku

$$\sigma_k(x) = \upsilon_k {c_k^2}^T x.$$

Distribuční funkce pro mez u_k je potom

$$P\{c_k^T x \geq u_k\} = 1 - F_k(\frac{u_k - {c_k^1}^T x}{{c_k^2}^T x}).$$

Pravděpodobnostní omezení

$$P\{c_k^T x \geq u_k\} \geq \beta_k, \quad \beta_k \in (0,1]$$

tak odpovídá nerovnosti

$$c_k^{1^T} x + F_k^{-1}(1-\beta_k) c_k^{2^T} x \geq u_k.$$

Tuto nerovnost přepíšeme do tvaru

$$\bar{f}_k(x) + \alpha_k \sigma_k(x) \geq u_k, \quad \alpha_k = \frac{F_k^{-1}(1-\beta_k) - \bar{t}_k}{\upsilon_k}.$$

Když $\alpha_k = \frac{F_k^{-1}(1-\beta_k)-\bar{t}_k}{\upsilon_k}$, tak potom $\alpha_k < (\geq)\, 0$, pokud $\beta_k > (\leq)\, 1 - F_k(\bar{t}_k)$.

U obou dvou případů vidíme, že pravděpodobnostní omezení

$$P\{f_k(x,\omega) \geq u_k\} \geq \beta_k, \quad \beta_k \in (0,1]$$

odpovídá nerovnosti

$$\bar{f}_k(x) + \alpha_k \sigma_k(x) \geq u_k.$$

Vidíme tedy, že se množina eficientních řešení $X_{K(\beta)}$ v obou případech shoduje s vícekriteriální úlohou

$$\max_{x \in X} [\bar{f}_1(x) + \alpha_1 \sigma_1(x), \ldots, \bar{f}_K(x) + \alpha_K \sigma_K(x)]. \tag{3.3}$$

Množinu eficientních řešení úlohy (3.3) označíme $X_{K\alpha}$.

Poznámka: S ohledem na konkávnost funkcí (viz první případ) a také na to, že je logické směrodatné odchylky minimalizovat, požadujeme zápornost koeficientů α.

Jestliže účelové funkce úlohy (3.2) splňují předpoklady z prvního nebo z druhého případu, pak platí:

Věta 3.2: [3] *Mějme zadané pravděpodobnosti $\beta_1, \ldots, \beta_K \in (0,1]$ tak, že s jejich pomocí vytvořené $\alpha_1, \ldots, \alpha_K$ (viz případy uvedené výše) jsou záporné. Pak množina eficientních řešení deterministické úlohy (3.3) s pravděpodobnostmi $\beta_1, \ldots, \beta_K$ je podmnožina množiny $X_{E\sigma}$.*

$$X_{K\alpha} \subset X_{E\sigma}$$

Poznámka: Ilustrace tohoto vztahu ve formě příkladu je uvedena rovněž v [3].

Když se v úloze (3.1) vyskytují náhodné parametry navíc i v omezeních $g(x,\omega)$ a rozhodnutí $x \in X$ je třeba zvolit dříve, než dojde k realizaci parametrů, nastává problém při určení množiny přípustných řešení X. Jak dojít k výsledku si nyní uvedeme.

3.2.2 Modely s pravděpodobnostními omezeními

Obecný tvar úlohy s pravděpodobnostními omezeními formulujeme takto:

$$\max E_P\left[z_1 f_1(x,\omega),\ldots,z_K f_K(x,\omega)\right] \tag{3.4}$$

$$\text{za podmínek } P\{g_j(x,\omega)\geq 0\}\geq \alpha_j,\ j=1,\ldots,m,\ x\in X,$$

kde $\alpha_j \in (0,1],\ j=1,\ldots,m$ jsou libovolná daná reálná čísla a $z_k : R \to R,\ k = 1,\ldots,K$ jsou vhodné funkce, jejichž střední hodnota existuje pro všechna $x \in X$.

Ideální by bylo, kdyby α_j pro všechna j bylo rovno 1. Omezení by tak bylo splněno skoro jistě a měli bychom tak vlastně množinu permanentně přípustných řešení. Takové omezení je však příliš veliké a taková množina by byla velice často prázdná.

Jak tuto překážku obejít je nasnadě – snížením hodnoty vektoru α. Tím sice snížíme požadovanou pravděpodobnost pro omezení (tudíž naše omezení nemusí být vždy splněno), ale zase si tím rozšíříme množinu přípustných řešení. Je třeba podotknout, že řešit tyto úlohy není vůbec snadné.

Pokud požadujeme, aby byly všechny podmínky splněny současně s určitou pravděpodobností, použijeme **sdružené pravděpodobnostní omezení** a úlohu můžeme formulovat jako

$$\max E_P\left[f_1(x,\omega),\ldots,f_K(x,\omega)\right]$$

$$\text{za podmínek } P\{g_j(x,\omega)\geq 0,\ j=1,\ldots,m\}\geq \alpha_0,\ x\in X,$$

kde $\alpha_0 \in (0,1]$ je libovolné dané reálné číslo.

Většinou je však lepší k jednotlivým omezením přistupovat individuálně. U některých je rozumné nastavit hranici vyšší, u jiných lze povolit i nižší. Tím jsme navedeni na myšlenku **individuálního pravděpodobnostního omezení**, kterou korektně zapíšeme jako

$$\max E_P\left[f_1(x,\omega),\ldots,f_K(x,\omega)\right]$$

$$\text{za podmínek } P\{g_j(x,\omega)\geq 0\}\geq \alpha_j,\ j=1,\ldots,m,\ x\in X,$$

kde $\alpha_j \in (0,1],\ j=1,\ldots,m$ jsou libovolná (většinou různá) reálná čísla.

3.2.3 Modely s penalizací ztrát

Uvažujme opět vícekriteriální optimalizační úlohu s náhodným elementem (3.1)

$$\max\left[f_1(x,\omega),\ldots,f_K(x,\omega)\right]$$

$$\text{za podmínek } g(x,\omega)\geq 0,\ x\in X.$$

Občas se setkáme se situací, že pro určitou realizaci ω^* vektoru ω nemusí platit $g(x,\omega^*) \geq 0$. Tím nám samozřejmě vzniká nějaká ztráta. Tuto ztrátu označme $\varphi(x,\omega)$. Předpokládáme, že $\varphi(x,\omega)$ je definovaná pro každé $x \in X$ a každé $\omega \in \Omega$.

$$\varphi(x,\omega) : R^n \times \Omega \to R^{m^*},$$

kde m^* je dáno charakterem uvažované penalizace.

Pokud známe $\varphi(x,\omega)$ explicitně, převede se úloha na případ náhody pouze v účelové funkci popisovaný v oddílu 3.2.1 a pokračujeme postupy uvedenými v tomto odstavci.

Uvažujme transformaci odpovídající E-kriteriu. Naše **úloha s penalizací ztrát** pak vypadá následujícím způsobem:

$$\max\ E_P\left[f_1(x,\omega) - \varphi(x,\omega), \ldots, f_K(x,\omega) - \varphi(x,\omega)\right]$$

$$\text{za podmínky } x \in X' \subset X,$$

kde množina X' je dána deterministickými omezeními pro existenci $\varphi(x,\omega)$, případně pro existenci středních hodnot $E_P\, f_k(x,\omega),\ k = 1, \ldots, K$ a $E_P\, \varphi(x,\omega)$ na množině X.

Nyní si uvedeme několik slov k volbě funce φ. Tato funkce se volí měřitelná, většinou nezáporná. Závisí na funkcích $g_j(x,\omega),\ j = 1, \ldots, m$ (nebo i na jiných náhodných parametrech). Bývá separovatelná vzhledem k jednotlivým funkcím $g_j(x,\omega)$, tj. celková ztráta z porušení více omezení je rovna součtu ztrát, které vzniknou z porušení těchto omezení jednotlivě. Logický je požadavek na nulovost ztráty, pokud $g_j(x,\omega) \geq 0,\ j = 1, \ldots, m$.

Příklad 3.3: Jednak pro ilustraci dosud zmíněných pojmů a také kvůli uvedení kompenzačních úloh si popíšeme následující (jednokriteriální) příklad. Jedná se o problém vaření v nemocnici.

Jídelna v nemocnici musí řešit každý den problém, kolik má uvařit jakých diet. Každý pacient pobývající v této nemocnici samozřejmě nemůže sníst všechno. Někdo má problémy se žaludkem, jiný se žlučníkem, další má cukrovku, nebo má rozbitou čelist a nemůže tak polykat. Protože jde o velkou nemocnici, kuchaři nemají přesné zprávy, kolik mají uvařit který den jakých jídel. Pokud je nějakých diet málo, musí je jídelna obstarat jinde, samozřejmě za vyšší cenu. Pokud je jiných diet na druhou stranu moc, za jejich likvidaci se také platí nějaké náklady. Snahou tedy je mít uvařené diety co nejpřesněji, aby penalizace za nepřesné počty diet byla co nejmenší.

V úloze se vyskytují proměnné

$p \in R^m$ ... pacienti v nemocnici, strávníci (i-tá složka vektoru p vyjadřuje počet pacientů s dietou i),

$d \in R^m$ ... počet vyrobených diet (i-tá složka vektoru d vyjadřuje počet diet typu i),

$a^0 \in R^m$ ... výrobní cena diet (i-tá složka vektoru a^0 vyjadřuje výrobní cenu jedné

diety typu i),
$a^< \in R^m$ ... cena chybějících diet zakoupených jinde, $a^< > a^0$ (i-tá složka vektoru $a^<$ vyjadřuje nákupní cenu jedné diety typu i),
$a^> \in R^m$ ... náklady spojené s likvidací přebytečných diet (i-tá složka vektoru $a^>$ vyjadřuje náklady spojené s likvidací jedné diety typu i),
a^0, $a^>$, $a^<$ jsou deterministické vektory, p a d jsou náhodné. Ty dopředu neznáme. Navíc je předpokládáme, že všechny uvedené proměnné nabývají nezáporných hodnot.

Cílem je minimalizovat náklady. Jelikož se tento problém řeší každý den, použijeme ve funkci střední hodnotu. Dostáváme výslednou úlohu

$$\min E_P\,[a^{0^T} d + a^{>^T}(p-d)^- + a^{<^T}(p-d)^+],$$

kde část $a^{>^T}(p-d)^- + a^{<^T}(p-d)^+$ vyjadřuje penalizaci za to, že obědy nejsou zajištěné přesně.

Tímto příkladem na penalizaci jsme si připravili půdu pro související jev – kompenzaci. Označíme $\eta = p - d$ rozdíl mezi vektory p a d. Kompenzujeme vektory p a d na rovnost. Děláme tak dodatečnou opravu, abychom se trefili do podmínek. η má (podle velikosti jednotlivých složek p a d) kladné i záporné složky (η^+, η^-).

Ztrátu, kterou chceme minimalizovat, vyjádříme jako

$$\varphi = \{a^< \eta^+ + a^> \eta^- \mid \eta^+ - \eta^- = p - d\}.$$

Pokud jsme schopni penalizační člen zapsat nějakou takovou minimalizací, jedná se o kompenzaci. Kompenzace je tedy vlastně jakousi podmnožinou penalizace.

Obecně problém lineární kompenzace popíšeme tímto způsobem. V prvním kroku zvolené rozhodnutí $x \in X$ nevyhovuje soustavě $T(\omega)x = h'(\omega)$. V dalším kroku tedy neshodu kompenzuji vektorem $y(x,\omega) \in R^{m^*}$. Tato úprava něco stojí. Náklady získám řešením tzv. úlohy druhého stupně

$$Q(x,\omega) = \min\,\{q^T(\omega)y \mid W(\omega)y = h'(\omega) - T(\omega)x,\ y \geq 0\},$$

kde $W \in R^{l^* \times m^*}$ je kompenzační matice, $T \in R^{l^* \times n}$ je technologická matice, vektor $h' \in R^{l^*}$ je pravá strana rovnice, většinou deterministický vektor $q \in R^{m^*}$ je cena úprav $y(x,\omega)$, může mít prvky určeny důležitostí jednotlivých podmínek. Samozřejmě ω označuje náhodu.

Celá úloha vypadá následovně:

$$\max_{x \in X} E_P\,[f_1(x,\omega) - Q(x,\omega), \ldots, f_K(x,\omega) - Q(x,\omega)]. \tag{3.5}$$

Pokud kompenzační matice W obsahuje náhodné prvky, jde o **úlohu s náhodnou kompenzací**.

Pokud je W deterministická, odpovídající úloze říkáme **úloha s pevnou kompenzací**.
Jejím speciálním případem je tzv. **úloha s jednoduchou kompenzací**. U ní platí

$$W = (I| - I), \text{ kde } I \in R^{l^* \times l^*} \text{ označuje jednotkovou matici.}$$

Pokud platí, že soustava $Wy = b'$ má řešení $y \geq 0$ pro všechna $b' \in R^{l*}$ (neboli jsme schopní zkompenzovat libovolný tvar pravé strany), mluvíme o **úloze s úplnou kompenzací**.

Úlohami s úplnou pevnou kompenzací se zabývá Cho v [11]. Jeho práce je však zaměřena na stabilitu.

Poznámka: Všechny zmiňované matice a vektory mají takové rozměry, aby zápis dával smysl.

3.2.4 Další případy

Redukce počtu účelových funkcí

V některých případech se může hodit trik uvedený v [14]. Část kriterií (účelových funkcí) se přesune mezi omezení a počet kriteriálních funkcí se tak zredukuje. Může se tím snížit náročnost výpočtů.

Předpokládejme, že opět řešíme úlohu vícekriteriálního stochastického programování (3.1). Původní počet K kriterií bychom potřebovali snížit na K^*, $K^* < K$. Bez újmy na obecnosti předpokládejme, že preferované účelové funkce $f_k(x, \omega)$ mají indexy $k = 1, \dots, K^*$ a "postradatelné" účelové funkce jsou oindexované $k = K^* + 1, \dots, K$. U těchto funkcí nám stačí, aby byly alespoň nějak dobré, pro nás přijatelné. Docílíme toho tak, že určíme meze $\delta_k \in R$, které by měly funkce $f_k(x, \omega)$ minimálně dosáhnout ($k = K^* + 1, \dots, K$). Problém (3.1) tedy tímto způsobem převedeme na úlohu

$$\max E_P \left[f_1(x, \omega), \dots, f_{K^*}(x, \omega) \right]$$

za podmínek

$$P\{f_k(x, \omega) \geq \delta_k\} \geq \beta_k, \quad k = K^* + 1, \dots, K,$$
$$P\{g_j(x, \omega) \geq 0\} \geq \beta_j, \quad j = 1, \dots, m,$$
$$x \in X.$$

Lze použít i sdružené pravděpodobnostní omezení a podmínky formulovat jako

$$P\{f_k(x, \omega) \geq \delta_k,\, k = K^* + 1, \dots, K;\ g_j(x, \omega) \geq 0,\, j = 1, \dots, m\} \geq \beta_0,$$

$$x \in X.$$

Volíme $\beta_k \in (0, 1]$, $k = K^* + 1, \dots, K$, $\beta_j \in (0, 1]$, $j = 1, \dots, m$, $\beta_0 \in (0, 1]$.

Neznalost rozdělení náhodných parametrů

Co si počít v případě, kdy nemáme informace o rozdělení náhodných veličin? V takové situaci si můžeme pomoci například empirickou distribucí (daty), kterou nahradíme skutečnou (ale neznámou) distribuční funkci. Potom zkoumáme vlastnosti odhadů příslušných charakteristik úlohy. S problematikou stochastických odhadů je úzce spjata otázka stability úloh stochastického programování. Ta se zabývá studiem změn optimálních (eficientních) řešení v závislosti na malých změnách v rozdělení. Těmto případům se věnují například V. Kaňková [14] a S. Vyvialová [18].

3.3 Příklady vícekriteriálních stochastických úloh

Optimální volba portfolia – Markowitzův model ([5])

Markowitzův model se zabývá optimálním složením portfolia. Investor má k dispozici n druhů neomezeně dělitelných cenných papírů (akcií), do kterých investuje svůj majetek. Jedná se o statický model bez transakčních nákladů, bez daní. Na druhou stranu nejsou povoleny krátké prodeje. Vyskytují se zde malí investoři, kteří investují pouze na jedno období. Model nepočítá s budoucností (uvažuje jen "od – do" a dál nic neřeší). Cílem investora je mít co největší zisk při co nejmenším riziku.

Prostředky vložené do akcie typu i v portfoliu označíme $x_i,\ i = 1, \dots, n$. Použití všech prostředků vyjádříme podmínkou

$$\sum_{i=1}^{n} x_i = 1. \tag{3.6}$$

Zákaz krátkých prodejů vyjádříme podmínkou

$$x_i \geq 0, \quad i = 1, \dots, n. \tag{3.7}$$

ρ_i nám označuje výnos z jednotkové investice do i-té akcie na konci zvoleného období. Jedná se o náhodnou veličinu.

Předpokládáme, že známe střední hodnoty $r_i = E\,\rho_i,\ i = 1, \dots, n$ (vektorově pro všechny druhy akcií $r = E\,\rho$) a rozptyly $v_{ii} = var(\rho_i)$, $v_{ii'} = cov(\rho_i, \rho_{i'}),\ i, i' = 1, \dots, n$ (varianční matice $V = var(\rho)$).

Požadujeme maximální zisk (výnos) vyjádřený střední hodnotou výnosnosti portfolia

$$r(x) = E\sum_{i=1}^{n} \rho_i x_i = \sum_{i=1}^{n} r_i x_i = r^T x \longrightarrow \max$$

a zároveň minimální riziko, zde vyjádřené rozptylem výnosnosti portfolia

$$\sigma^2(x) = E(\sum_{i=1}^{n} \rho_i x_i - \sum_{i=1}^{n} r_i x_i)^2 = \sum_{i=1}^{n}\sum_{i'=1}^{n} x_i v_{ii'} x_{i'} = x^T V x \longrightarrow \min$$

za podmínek (3.6), (3.7).

Investor se tedy zabývá problémem

$$\max\,[r^T x,\ -x^T V x] \text{ za podmínek } (3.6), (3.7).$$

"Optimální" řešení této úlohy získáme různými způsoby. Uvedeme si tři možnosti.

1. $$\max\, k_1 r^T x - k_2 x^T V x$$

   za podmínek (3.6), (3.7), $k_1,\, k_2 > 0$.
   Čím je k_1 větší, tím je investice riskantnější.

2. $$\max\, r^T x$$

   za podmínek (3.6), (3.7) a $x^T V x \leq m_1$, kde m_1 je daný parametr.

3. $$\min\, x^T V x$$

   za podmínek (3.6), (3.7) a $r^T x \geq m_2$, kde m_2 je daný parametr.

Matematický model pro produkci kovové slitiny ([16])

Úkolem je nalézt optimální postup k získání určité slitiny. K dispozici máme m různých surovin. Na kvalitě směsi se podílí vhodný poměr z n různých chemických prvků, které se v jednotlivých surovinách vyskytují.

Označíme

$x_i \ldots$ množství suroviny i (v kg) v jedné tuně směsi,

$c_i^* \ldots$ cena jednoho kilogramu suroviny i,

$a_{ij} \ldots$ množství chemického prvku j v surovině i (v procentech),

$a_i \ldots$ maximální množství suroviny i (v kg), které může přijít do výsledné slitiny,

$p_j^{max} \ldots$ maximální množství (v procentech), ve kterém se chemický prvek j musí nacházet ve výsledné slitině,

$p_j^{min} \ldots$ minimální množství (v procentech), ve kterém se chemický prvek j musí nacházet ve výsledné slitině,

$z_j \ldots$ variační koeficient chemického prvku j, který popisuje změny u prvků, k nimž dochází kvůli technologickým procesům během tavení.

Vždy platí $i = 1, \ldots, m$ a $j = 1, \ldots, n$.

Požadujeme minimální náklady při výrobě slitiny a také musíme počítat s omezením na množství suroviny k. Ve výsledné směsi se má surovina k vyskytovat v co nejmenším množství. Může to být způsobeno například nedostatkem této suroviny, nebo její abnormálně vysokou cenou. Matematický model pro získání jedné tuny slitiny tedy zapíšeme jako

$$\min\,[\sum_{i=1}^{m} c_i^* x_i,\ x_k]$$

za podmínek

$$\sum_{i=1}^{m} x_i = 1000,$$

$$10p_j^{min} \le (1+\frac{z_j}{100})\sum_{i=1}^{m}\frac{a_{ij}}{100}x_i \le 10p_j^{max}, \quad j = 1,\dots,n, \tag{3.8}$$

$$0 \le x_i \le a_i, \quad i = 1,\dots,m.$$

Bude vhodné vysvětlit podmínku (3.8). Část $\sum\limits_{i=1}^{m}\frac{a_{ij}}{100}x_i$ říká, kolik je teoreticky celkově prvku j ve všech použitých surovinách ve slitině (v kg). Člen se z_j upravuje toto množství po tavení. Výsledné množství se musí vyskytovat pouze v omezeném množství. Meze p_j^{min} a p_j^{max} jsou udány v procentech, slitiny celkem má být jedna tuna, množství chemických prvků je v kg, proto jsou meze vynásobeny číslem 10.

Koeficienty z_j jsou kvůli rozmanitým technologickým procesům během tavení náhodné, takže budeme požadovat, aby podmínka (3.8) byla splněna s určitou pravděpodobností. Tuto podmínku tedy zapíšeme jako

$$P\{10p_j^{min} \le (1+\frac{z_j}{100})\sum_{i=1}^{m}\frac{a_{ij}}{100}x_i \le 10p_j^{max}\} \ge \dot{\delta}_j, \quad j = 1,\dots,n,$$

kde hodnoty všech $\dot{\delta}_j \in (0,1]$ můžeme znát například z praxe. Ostatní koeficienty jsou deterministické. Finální podoba této úlohy tedy je

$$\min\,[\sum_{i=1}^{m} c_i^* x_i,\ x_k]$$

za podmínek

$$\sum_{i=1}^{m} x_i = 1000,$$

$$P\{10p_j^{min} \le (1+\frac{z_j}{100})\sum_{i=1}^{m}\frac{a_{ij}}{100}x_i \le 10p_j^{max}\} \ge \dot{\delta}_j, \quad j = 1,\dots,n,$$

$$0 \le x_i \le a_i, \quad i = 1,\dots,m.$$

Problém rozdělení práce ([16])

V dílně pracuje n dělníků. Je potřeba splnit n úkolů. Každý pracovník bude dělat právě jednu práci a naším cílem je vhodně přiřadit každému dělníkovi tu co nejvíc vyhovující, aby splnění úkolů bylo co nejefektivnější. Typy prací jsou totiž odlišné. Také dělníci nejen že nemají stejnou výkonnost, ale jsou různě specializováni. Proto by třeba stejnou práci dělal každý z nich různě dlouho, s různými náklady, spotřebou materiálu a různou produktivitou. Samozřejmě každý z dělníků by více preferoval takové práce, ke kterým je specializován.

Zadefinujeme si proměnnou x_{ij}, která nabývá pouze hodnot 0 a 1. Tato proměnná indikuje, zda je pracovníkovi K_i přidělena práce L_j, nebo nikoliv.

$$x_{ij} = \begin{cases} 1, & \text{pracovníkovi } K_i \text{ je přidělena práce } L_j \\ 0, & \text{jinak} \end{cases}$$

Dále označíme

$t_{ij} \ldots$ čas, za jak dlouho dokončí dělník K_i práci L_j,

$c^*_{ij} \ldots$ náklady na vykonání práce L_j dělníkem K_i,

$p_{ij} \ldots$ produktivita dělníka K_i při práci L_j (vyjádřená jako počet zpracovaných položek za jednotku času),

$cs_{ij} \ldots$ spotřeba vzácných (nedostatkových) materiálů dělníka K_i při práci L_j,

$csn_j \ldots$ normovaná spotřeba vzácných materiálů při práci L_j,

$\frac{cs_{ij}}{csn_j} \ldots$ index specifické spotřeby (dělníka K_i při práci L_j),

$p^*_{ij} \ldots$ preference dělníka K_i práce L_j.

Vždy platí $i = 1, \ldots, n$ a $j = 1, \ldots, n$.

Všechny uvedené veličiny jsou náhodné. Čas potřebný k dokončení určité práce závisí na spoustě okolností. Například na motivaci a náladě pracovníka, na stavu využívaných strojů. Tím pádem jsou náhodné i náklady s produktivitou, protože čím déle bude nějaká práce trvat, tím budou větší náklady a menší produktivita. Náklady jsou také samozřejmě ovlivněny kvalifikací dělníků. Čím bude kvalifikace dělníků vyšší, tím budou vyšší i náklady. Preference dělníků se také mohou lišit. Někdy se zaměstnanec nemusí cítit dobře, nebo potřebuje skončit dřív, tak by radši dělal nějakou lehčí a méně časově náročnou práci. Existuje ještě mnoho dalších podobných náhodných faktorů ovlivňujících zavedené proměnné.

Jak již bylo řečeno, naším cílem je přidělit co nejefektivněji každému dělníkovi právě jednu práci. Naší snahou přitom je minimalizovat celkový čas k provedení všech prací

$$f_1 = E \sum_{i=1}^{n} \sum_{j=1}^{n} t_{ij} x_{ij},$$

minimalizovat celkové náklady

$$f_2 = E \sum_{i=1}^{n} \sum_{j=1}^{n} c^*_{ij} x_{ij},$$

dosáhnout co nejvyšší celkové produktivity práce

$$f_3 = E \sum_{i=1}^{n} \sum_{j=1}^{n} p_{ij} x_{ij},$$

co nejvíce snížit index specifické spotřeby

$$f_4 = E \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{cs_{ij}}{csn_j} x_{ij}$$

a pokusit se o maximální splnění preferencí dělníků

$$f_5 = E \sum_{i=1}^{n} \sum_{j=1}^{n} p^*_{ij} x_{ij}.$$

Matematický model pro tuto úlohu tedy formulujeme jako

$$\max \left[-f_1,\ -f_2,\ f_3,\ -f_4,\ f_5\right]$$

za podmínek

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

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

$$x_{ij} \in \{0, 1\}, \quad i, j = 1, \dots, n.$$

Efektivní využití kapacity strojů ([16])

Tento příklad se zabývá zefektivněním výrobního procesu v továrně, jejíž hlavní činnost spočívá ve svařování. K dispozici je určitý omezený počet přístrojů a cílem je s jejich pomocí vyrobit co nejvíce kusů různých výrobků alespoň tak, aby byl splněn měsíční plán.

Zadefinujeme si proměnné
x_{ij}... počet výrobků typu i zpracovaných přístrojem j,

$$I_{ij} = \begin{cases} 1, & \text{přístroj } j \text{ se podílí na produkci výrobku } i \\ 0, & \text{jinak.} \end{cases}$$

Dále označíme

M_V... množina typů výrobků, které se v továrně vyrábí,

M_P... množina přístrojů, které jsou v továrně k dispozici,

$T_i,\ i \in M_V$... čas potřebný k výrobě jednoho výrobku typu i,

$N_i,\ i \in M_V$... požadované minimální množství výrobků i vyrobené za jeden měsíc,

$C_j,\ j \in M_P \ldots$ měsíční časová kapacita přístroje j,
$\bar{s}_j,\ j \in M_P \ldots$ střední čas vytížení přístroje j.

Cílem vedení továrny je dosáhnout co nejdelšího času pro výrobu u jednotlivých přístrojů (tedy zpracovat co nejvíce výrobků)

$$\max f_1 = \sum_{i \in M_V} \sum_{j \in M_P} T_i I_{ij} x_{ij}$$

a pokusit se o co nejpodobnější zatížení používaných strojů

$$\min f_2 = \sum_{j \in M_P} (\sum_{i \in M_V} T_i I_{ij} x_{ij} - \bar{s}_j)^2.$$

Pochopitelně musí být splněny podmínky

$$\sum_{j \in M_P} I_{ij} x_{ij} \geq N_i, \quad i \in M_V, \tag{3.9}$$

$$\sum_{i \in M_V} T_i I_{ij} x_{ij} \leq C_j, \quad j \in M_P, \tag{3.10}$$

$$x_{ij} \geq 0, \quad i \in M_V,\ j \in M_P.$$

Omezení (3.9) hlídá splnění minimální výroby. Výrobků může být vyrobeno i více než se požaduje, protože případné přebytečné se mohou umístit do skladu, nebo je možné, že si je ještě někdo přiobjedná až po spuštění výroby.

Omezení (3.10) se týká maximálního zatížení přístrojů během měsíce. Mez C_j by u přístroje j neměla být překročena. Jinak hrozí například porucha.

V této úloze jsou T_i náhodné veličiny. Vzhledem k tomu, že se mohou lišit doby zatížení během jednotlivých měsíců, jsou náhodné i C_j. Podmínku (3.10) tak lze nahradit podmínkou

$$P\{\sum_{i \in M_V} T_i I_{ij} x_{ij} \leq C_j\} \geq \ddot{\delta}_j, \quad j \in M_P,$$

když je ozkoušeno z praxe, že hodnoty $\ddot{\delta}_j \in (0, 1]$ lze použít.

Náhodu v účelových funkcích je možné odstranit například pomocí dvou mezí $\tilde{f}_1$ a $\tilde{f}_2$ určených tak, že pokud by funkce f_1 dosahovala alespoň hodnoty $\tilde{f}_1$ a funkce f_2 by nepřesahovala $\tilde{f}_2$, panovala by s produkcí spokojenost. Výslednou úlohu pak lze zapsat jako

$$\max P\{f_1 \geq \tilde{f}_1,\ f_2 \leq \tilde{f}_2\}$$

za podmínek

$$\sum_{j \in M_P} I_{ij} x_{ij} \geq N_i, \quad i \in M_V,$$

$$P\{\sum_{i \in M_V} T_i I_{ij} x_{ij} \leq C_j\} \geq \ddot{\delta}_j, \quad j \in M_P,$$

$$x_{ij} \geq 0, \quad i \in M_V,\ j \in M_P.$$

Poznámka: V tomto příkladu se setkáváme s kombinací P-kriteria a modelů s pravděpodobnostními omezeními.

Kapitola 4

Ilustrační příklad

Nyní si ukážeme jednu konkrétní aplikaci ze zmiňovaných postupů. Budeme se věnovat příkladu optimálního složení portfolia pomocí Markowitzova modelu. K numerické ilustraci využíváme reálná data.

4.1 Optimální složení portfolia

Naším úkolem je investovat určitý majetek do akcií za nějakých podmínek a omezení. Samozřejmě si logicky přejeme, abychom z této investice měli co největší zisk s co nejmenším rizikem. K optimálnímu výběru akcií do portfolia použijeme princip již probíraného Markowitzova modelu.

Máme k dispozici seznam n druhů akcií. Chceme investovat na pevně stanovené období a nic víc nás nezajímá. Předpokládáme, že akcie jsou neomezeně dělitelné a v rámci statického modelu neuvažujeme žádné transakční náklady a žádné daně. Je potřeba rozlišovat, zda jsou povoleny krátké prodeje (*short sales allowed*), či nikoliv. Pokud povoleny budou, opět neuvažujeme žádné další poplatky a budeme předpokládat, že všechny prováděné operace lze uskutečnit.

Výnos z jednotkové investice do i-té akcie na konci zvoleného období označíme ρ_i. Hodnoty těchto výnosů však dopředu neznáme. Jedná se o náhodné veličiny.

Předpokládáme, že známe střední hodnoty $r_i = E\,\rho_i,\ i = 1, \ldots, n$ (vektorově pro všechny druhy akcií $r = E\,\rho$) a rozptyly $v_{ii} = var(\rho_i)$, $v_{ii'} = cov(\rho_i, \rho_{i'}),\ i,\ i' = 1, \ldots, n$ (varianční matice $V = var(\rho)$).

Rovněž předpokládáme, že jednotlivé hodnoty r_i nejsou stejné (vektor r a jednotkový vektor stejné dimenze jsou lineárně nezávislé) a matice V je pozitivně definitní (vylučuje bezrizikovost).

Především chceme maximalizovat zisk (výnos) z naší investice. Vyjádříme ho funkcí rx. Množina přípustných řešení X má základní tvar

$$X = \{x \in R^n \mid \sum_{i=1}^{n} x_i = 1\}.$$

Může být však zúžena dalšími podmínkami, jako například $x_i \geq 0,\ i = 1, \ldots, n$ (nelze použít krátkodobé dluhopisy), $x_i \leq const,\ i = 1, \ldots, n$, kde $const \in (0, 1)$ označuje maximální přípustnou investici do jednoho druhu akcií. Toto omezení lze využít, když chceme z nějakého důvodu zamezit investování většiny majetku do jediného titulu.

Chceme-li maximální výnos rx, je potřeba vložit veškerý majetek do akcií s nejvyšší výnosností. Taková operace je však velmi riskantní. Z mnoha různých příčin můžeme takovou investicí prodělat. Do našich výpočtů je vhodné zahrnout i riziko spojené s investováním. Toto riziko je vyjádřené funkcí $x^T V x$ (rozptyl výnosu portfolia), kterou se snažíme samozřejmě minimalizovat. V rámci sestavení optimálního portfolia lze použít i jiná rizika, jako například směrodatnou odchylku výnosu portfolia $\sqrt{x^T V x}$.

Nejméně riskantní řešení (řešení úlohy min $x^T V x$ na $X = \{x \in R^n \mid \sum_{i=1}^n x_i = 1\}$) zjistíme pomocí Lagrangeovy funkce.

$$L(x, \mu) = x^T V x + \mu(\mathbf{1}^T x - 1)$$

$$2Vx + \mu\mathbf{1} = 0$$

$$x = -\frac{\mu}{2} V^{-1}\mathbf{1}$$

μ vypočítáme díky podmínce $\mathbf{1}^T x = 1$ a vynásobením poslední rovnosti $\mathbf{1}^T$.

$$1 = -\frac{\mu}{2}\mathbf{1}^T V^{-1}\mathbf{1}$$

$$\mu = -\frac{2}{\mathbf{1}^T V^{-1}\mathbf{1}}$$

Získali jsme tak řešení

$$x_M = \frac{V^{-1}\mathbf{1}}{\mathbf{1}^T V^{-1}\mathbf{1}}.$$

Toto řešení minimalizuje rozptyl $x^T V x$ bez ohledu na výnosnost. Není to ideální řešení, ale víme, že nemáme jít na výnos horší než je výnos pro x_M.
Minimální rozptyl je tedy

$$x_M^T V x_M = \frac{1}{\mathbf{1}^T V^{-1}\mathbf{1}}.$$

Výnos pro x_M je

$$r^T x_M = \frac{r^T V^{-1}\mathbf{1}}{\mathbf{1}^T V^{-1}\mathbf{1}}.$$

Pokud si klademe podmínku na výnos (požadujeme, aby předpokládaný výnos dosahoval výše $r_p \in R$ za minimálního rizika), opět použijeme Lagrangeovu funkci s multiplikátory μ_1, μ_2. Postupujeme podobně, jako při předchozím výpočtu nejméně riskantního řešení x_M.

$$L(x, \mu_1, \mu_2) = x^T V x + \mu_1(\mathbf{1}^T x - 1) + \mu_2(r^T x - r_p)$$

$$2Vx + \mu_1\mathbf{1} + \mu_2 r = 0$$

$$x = -\frac{1}{2}\mu_1 V^{-1}\mathbf{1} - \frac{1}{2}\mu_2 V^{-1} r$$

Tuto rovnici vynásobíme postupně $\mathbf{1}^T$ a r^T. Podobně jako v předchozím případě vypočítáme díky podmínkám $\mathbf{1}^T x = 1$ a $r^T x = r_p$ hodnoty μ_1 a μ_2 ze vzniklé soustavy dvou rovnic

$$1 = -\frac{1}{2}\mu_1 \mathbf{1}^T V^{-1}\mathbf{1} - \frac{1}{2}\mu_2 \mathbf{1}^T V^{-1} r$$

$$r_p = -\frac{1}{2}\mu_1 r^T V^{-1}\mathbf{1} - \frac{1}{2}\mu_2 r^T V^{-1} r$$

Po dosazení multiplikátorů μ_1 a μ_2 dostáváme požadované řešení x.

Poznámka: Tučná **1** označuje n-rozměrný jednotkový vektor.

Podobným způsobem lze postupovat i při výpočtu úlohy
max $r^T x$ za podmínky $x^T V x = \sigma_p^2$ na množině $X = \{x \in R^n \mid \sum_{i=1}^n x_i = 1\}$.

Abychom však neopustili prostor vícekriteriálních úloh, vrátíme se k úloze současné maximalizace výnosu $r^T x$ a minimalizace rizika $x^T V x$. Úlohu zapíšeme pomocí nezáporných vah k_1, k_2 jako

$$\max_{x \in X} k_1 r^T x - k_2 x^T V x.$$

Parametr k_1 vyjadřuje sklon investora k riziku. Čím je jeho hodnota vyšší, tím je investice riskantnější. Pokud se rizika bojíme, hodnota k_1 se blíží k 0. Pro naši numerickou ilustraci použijeme upravené parametry $k_1 = \lambda$ a $k_2 = (1 - \lambda)$, kde $0 \leq \lambda \leq 1$. Finální podoba naší úlohy tedy je

$$\max_{x \in X} \lambda r^T x - (1 - \lambda) x^T V x.$$

Seznámili jsme se se základními myšlenkami a principy úlohy optimálního složení portfolia pomocí Markowitzova modelu. Nyní si předvedeme konkrétní numerické výsledky. Využíváme reálná data z burzy přístupná na webu finance.yahoo.com.

4.2 Numerické výpočty a grafy

Uvažujme následující zadání úlohy. Máme za úkol sestavit optimální portfolio z 24 druhů akcií, které jsme si předem vybrali. Výběr titulů byl proveden tříděním akcií na adrese http://screen.yahoo.com/stocks.html. Požadovány byly ziskové tituly s větším objemem obchodů, předpokládaným růstem ceny, doporučené nyní ke koupi. Nakonec se vybralo 24 druhů akcií uvedených v tabulce 4.1.

K dispozici máme ceny akcií z minulosti, informace o štěpení a výplatě dividend u jednotlivých titulů. Pro určení odhadu středního výnosu akcie se použily pondělní

	zkratka	celý název	odvětví
x_1	DNB	Dun & Bradstreet Corp.	Information & Delivery Services
x_2	JHX	James Hardie Industries NV	Cement
x_3	NDAQ	NASDAQ Stock Market, Inc.	Business Services
x_4	NTRI	NutriSystem Inc.	Business Services
x_5	JNC	Nuveen Investments Inc.	Asset Management
x_6	TKC	Turkcell Iletisim Hizmetleri AS	Wireless Communications
x_7	TXU	TXU Corp.	Electric Utilities (Utilities)
x_8	AWC	Alumina Ltd.	Synthetics (Basic Materials)
x_9	UIC	United Industrial Corp.	Aerospace/Defense Products & Serv.
x_{10}	CHB	Champion Enterprises Inc.	Manufactured Housing
x_{11}	GMXR	GMX Resources Inc.	Independent Oil & Gas
x_{12}	RAVN	Raven Industries Inc.	Printed Circuit Boards
x_{13}	WLT	Walter Industries Inc.	Residential Construction
x_{14}	WOS	Wolseley plc	Diversified Machinery
x_{15}	AWGI	Alderwoods Group Inc.	Personal Services
x_{16}	ACO	Amcol International Corp.	General Building Materials
x_{17}	UHAL	AMERCO	Rental & Leasing Services
x_{18}	ARJ	Arch Chemicals Inc.	Synthetics
x_{19}	NPO	EnPro Industries Inc.	Aerospace/Defense
x_{20}	ESCL	ESCALA GROUP INC	Business Services
x_{21}	HHGP	Hudson Highland Group Inc.	Staffing & Outsourcing Services
x_{22}	MC	Matsushita Electric Industrial	Electronic Equipment
x_{23}	RTI	RTI International Metals Inc.	Industrial Metals & Minerals
x_{24}	CVNS	Covansys Corp.	Information Technology Services

Tabulka 4.1: Seznam vybraných akcií.

ceny této akcie (v dolarech) za poslední dva roky, konkrétně v období mezi 13. říjnem 2003 a 17. říjnem 2005. Jedná se o 106 hodnot. Před všemi výpočty se musel brát ohled na štěpení akcií (týká se TKC, RAVN, WOS, HHGP) a výplatu dividend (JHX, JNC, TKC, TXU, AWC, UIC, RAVN, WLT, WOS, ACO, UHAL, ARJ, MC). U inkriminovaných akcií se musely provést sjednocující operace. Výplata dividend se zahrnula pouze do odhadu středního výnosu akcie, do odhadu rozptylu výnosnosti nikoliv.

Označíme
$r_{it}\dots$ výnosnost i-té akcie za týden t (bez dividend),
$r_{it}^D\dots$ výnosnost i-té akcie za týden t (s dividendami),
$r_i\dots$ odhad středního výnosu akcie i (bez započítání dividend),
$r_i^D\dots$ odhad středního výnosu akcie i (se započítáním dividend),
$r = (r_i^D)_{i=1}^{24}$,
$V = (v_{ij})_{i,j=1}^{24}\dots$ odhad rozptylu výnosnosti,

$x_i \dots$ poměrné zastoupení akcie i v portfoliu, $\sum\limits_{i=1}^{24} x_i = 1$

$x = (x_i)_{i=1}^{24}$.
Vždy platí $t = 1, \dots, 105$, $i = 1, \dots, 24$.

Týdenní výnosnosti jednotlivých titulů spočítáme jako

$$r_{it} = \frac{\text{cena akcie}\, i \text{ v pondělí týdne } t}{\text{cena akcie}\, i \text{ v pondělí týdne } (t-1)} - 1,$$

$$t = 1, \dots, 105,\ i = 1, \dots, 24.$$

Pokud mezi týdny t a $(t-1)$ byly vyplaceny dividendy,

$$r_{it}^D = \frac{\text{cena akcie}\, i \text{ v pondělí týdne } t + dividendy}{\text{cena akcie}\, i \text{ v pondělí týdne } (t-1)} - 1,$$

jinak

$$r_{it}^D = r_{it},$$

$$t = 1, \dots, 105,\ i = 1, \dots, 24.$$

Odhad středního výnosu akcií je

$$r_i = \frac{1}{105} \sum_{t=1}^{105} r_{it}, \quad i = 1, \dots, 24,$$

$$r_i^D = \frac{1}{105} \sum_{t=1}^{105} r_{it}^D, \quad i = 1, \dots, 24.$$

Odhad rozptylu výnosnosti je

$$v_{ij} = \frac{1}{104} \sum_{t=1}^{105} (r_{it} - r_i)(r_{jt} - r_j),$$

$$i = 1, \dots, 24,\ j = 1, \dots, 24.$$

Všechny výpočty se prováděly v tabulkovém procesoru EXCEL. Konkrétní hodnoty odhadů středních výnosů r a rozptylů výnosnosti V jsou uvedeny v příloze. Empirické odhady Markowitzova modelu byly studovány v [13].

Teď, když známe hodnoty r a V, můžeme přistoupit k samotné optimalizaci. Jak již bylo několikrát řečeno, chceme maximalizovat střední výnos portfolia $r^T x$ a minimalizovat rozptyl výnosu portfolia $x^T V x$. Toto vede na úlohu vícekriteriální optimalizace

$$\max_{x \in X} \lambda r^T x - (1 - \lambda) x^T V x. \tag{4.1}$$

Ukážeme si, jak se výsledky úlohy mění v závislosti na zvoleném parametru λ. Jeho hodnoty se postupně zvyšovaly po dílcích o velikosti 0,05 od 0 až do 1 včetně.

Pro přesnější graf znázorňující závislost očekávané výnosnosti na rozptylu výnosnosti (tzv. efektivní hranice portfolia) byly dílky pouze setinové.

Velký vliv mají samozřejmě různě zvolené množiny přípustných řešení. Záleží, zda jsou povoleny krátké prodeje (a do jaké výše počátečního vkladu), či nikoliv. Další použité omezení je požadavek na maximální podíl investovaného majetku do jednotlivých titulů.

Jako jiný druh rizika se místo $x^T V x$ použila směrodatná odchylka výnosu portfolia $\sqrt{x^T V x}$. Pozměněná úloha pak má tvar

$$\max_{x \in X} \lambda r^T x - (1 - \lambda)\sqrt{x^T V x}. \tag{4.2}$$

Jelikož všech výsledných dat této úlohy je velmi mnoho (díky velkým počtům hodnot λ a titulů k investování), jejich konkrétní podobu si zde uvádět nebudeme. Kompletní výsledky pro všech 21 hodnot λ mezi 0 a 1 jsou k dispozici v příloze.

Graficky si znázorníme závislosti očekávané výnosnosti a rozptylu výnosnosti (resp. směrodatné odchylky výnosnosti) na parametru λ. Jednotlivé body na obrázku mají souřadnice $[\lambda, r^T x]$ v případě výnosnosti, $[\lambda, x^T V x]$ v případě rozptylu výnosnosti a $[\lambda, \sqrt{x^T V x}]$ v případě směrodatné odchylky výnosnosti. Vektory x získáme vyřešením úlohy (4.1) (resp. úlohy (4.2)) s příslušnou hodnotou λ.

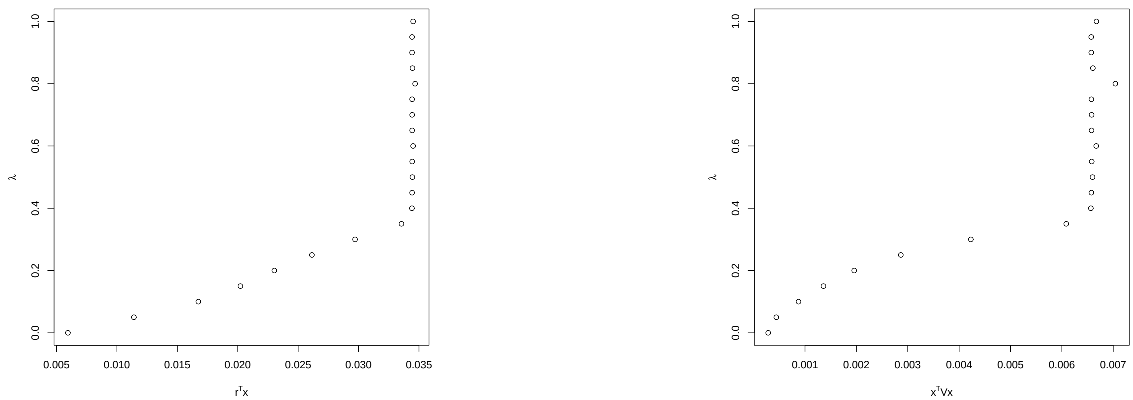
Dále si předvedeme závislost očekávané výnosnosti na rozptylu výnosnosti (resp. na její směrodatné odchylce). Tuto efektivní hranici portfolia ukážeme i na podrobnějším grafu s větší hustotou λ (dílky o velikosti 0,01). Body na obrázku mají souřadnice $[r^T x, x^T V x]$ (resp. $[r^T x, \sqrt{x^T V x}]$). Jednotlivá x získáme opět vyřešením úlohy (4.1) (resp. úlohy (4.2)) s příslušnou hodnotou λ.

Pomocí těchto všech grafů tedy lze pozorovat změny očekávaných výnosů a rizik v závislosti na hodnotě λ. Snadno si můžeme zvolit nám vyhovující λ a tím pádem řešení x, které zajistí požadovaný výnos s odpovídajícím rizikem.

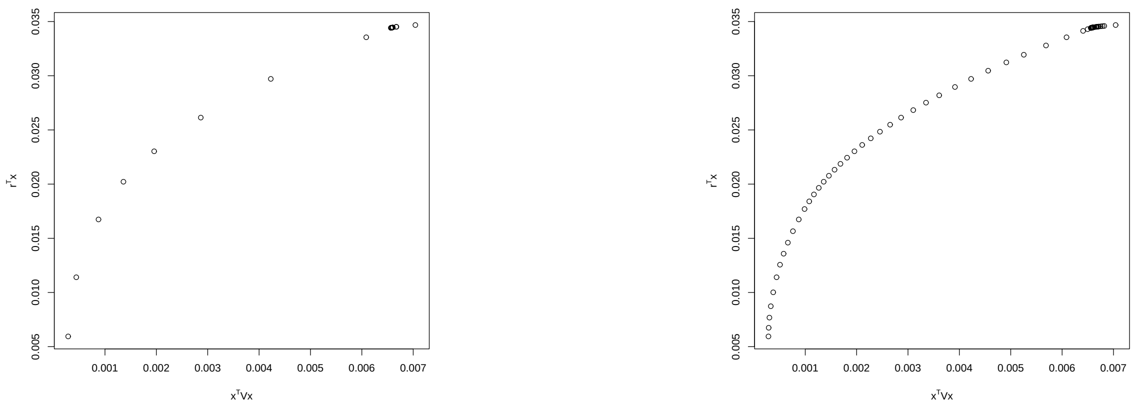
4.2.1 Úloha (4.1) - riziko $x^T V x$

A) $X = \{x \in R^{24} \mid \sum_{i=1}^{24} x_i = 1,\ x_i \geq 0\ \forall i\}$

Omezení na množinu přípustných řešení $x_i \geq 0\ \forall i$ znamená, že nejsou povoleny krátké prodeje.



Obrázek 4.1: Závislost očekávaného výnosu a rozptylu výnosnosti na λ.

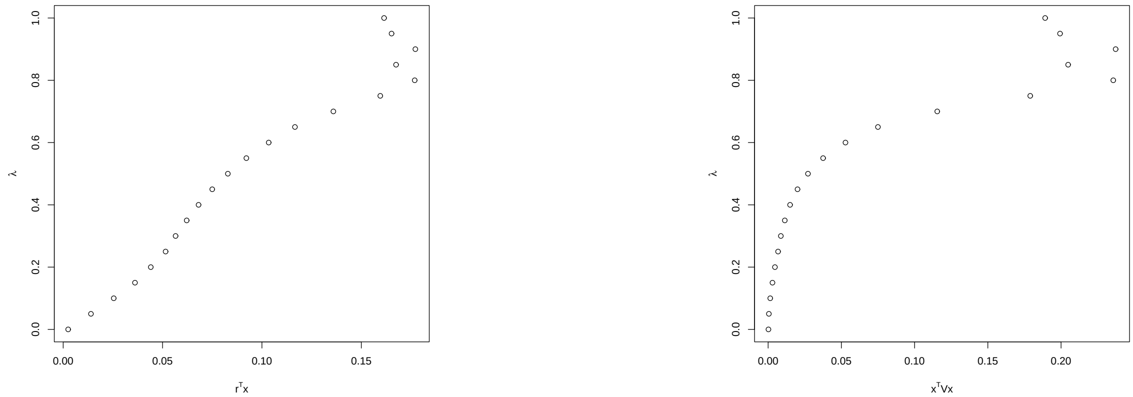


Obrázek 4.2: Efektivní hranice portfolia (21 a 101 pozorování).

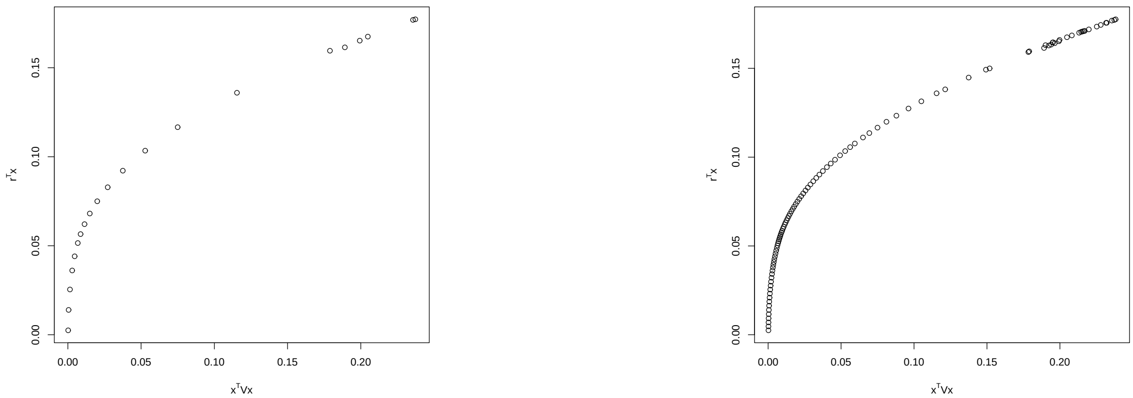
B) $X = \{x \in R^{24} \mid \sum_{i=1}^{24} x_i = 1,\ x_i \geq -konst_1\ \forall i\}$

Omezení $x_i \geq -konst_1\ \forall i$ znamená, že jsou povoleny krátké prodeje až do výše $100 konst_1\%$ vkladu.

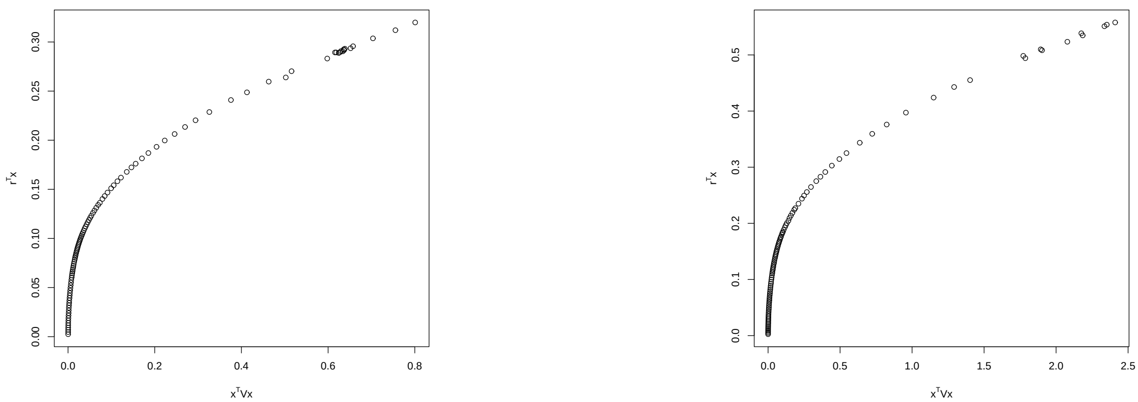
Představíme si především grafické výstupy pro $konst_1 = 0{,}25$. Jsou tedy povoleny krátké prodeje do výše 25%. Znázorníme si i efektivní hranici portfolia pro $konst_1 = 0{,}50$ a $konst_1 = 1$.



Obrázek 4.3: Závislost očekávaného výnosu a rozptylu výnosnosti na λ.



Obrázek 4.4: Efektivní hranice portfolia (21 a 101 pozorování, $konst_1 = 0{,}25$).

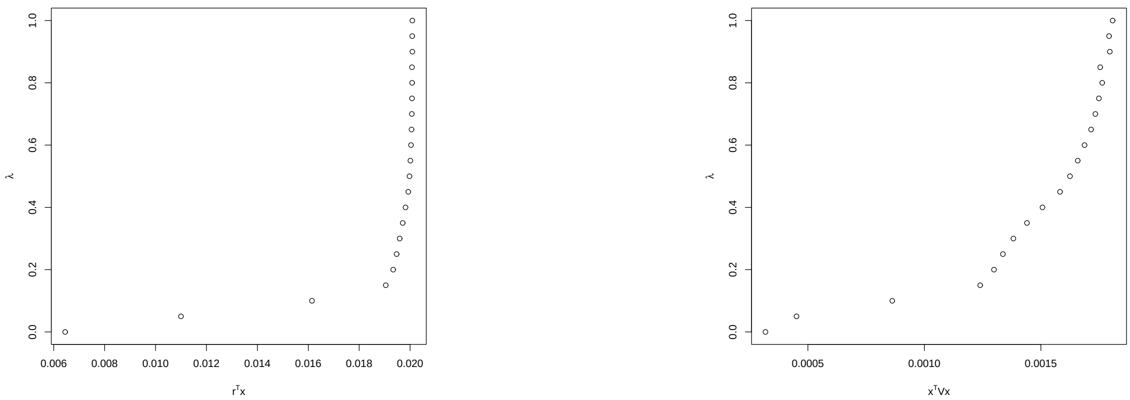


Obrázek 4.5: Efektivní hranice portfolia (101 pozorování, $konst_1 = 0{,}5$ a $konst_1 = 1$).

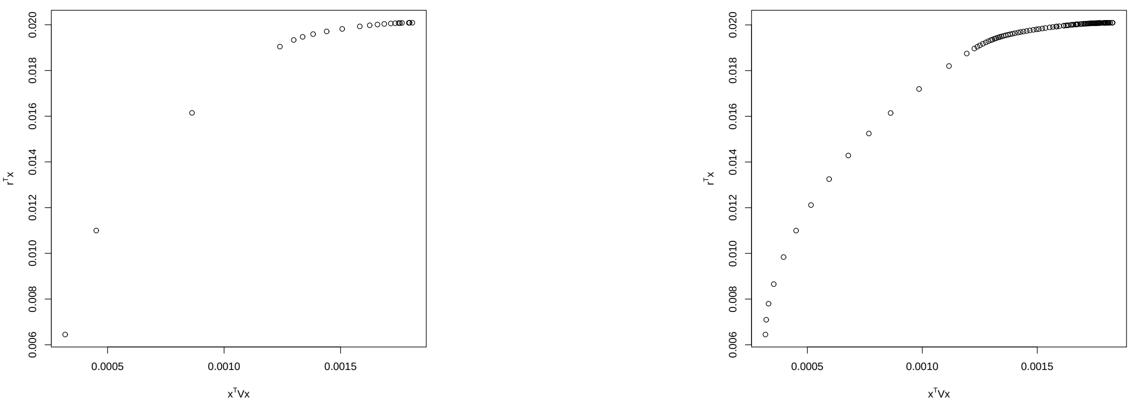
C) $X = \{x \in R^{24} \mid \sum_{i=1}^{24} x_i = 1,\ x_i \geq 0\ \forall i,\ x_i \leq konst_2\ \forall i\}$

Zde již nejsou povoleny krátké prodeje, navíc podmínka $x_i \leq konst_2\ \forall i$ vyjadřuje omezení, že žádný z titulů nesmí přesáhnout váhu $100 konst_2\%$ v celkovém portfoliu.

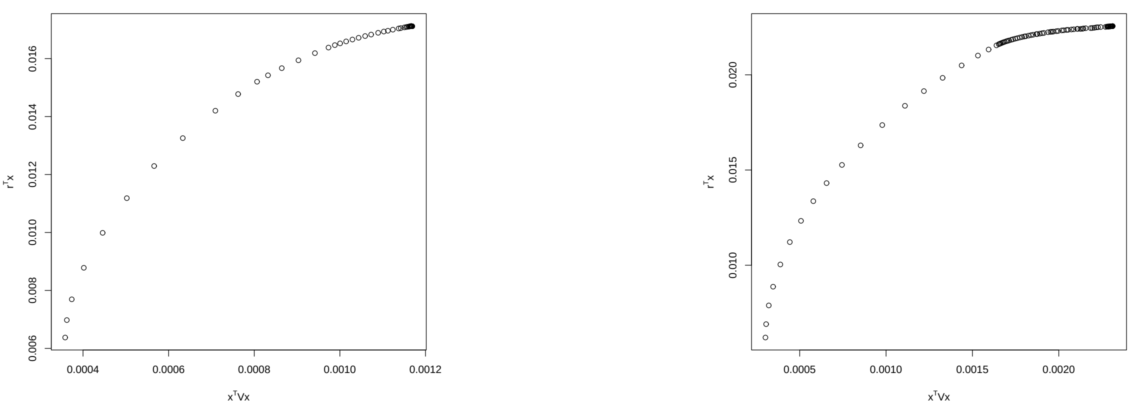
Ukážeme si grafické výstupy pro $konst_2 = 0{,}15$. Do žádného titulu tedy nesmíme investovat více než 15% z našeho majetku. Předvedeme si i závislost očekávané výnosnosti na rozptylu výnosnosti pro $konst_2 = 0{,}10$ a $konst_2 = 0{,}20$.



Obrázek 4.6: Závislost očekávaného výnosu a rozptylu výnosnosti na λ.



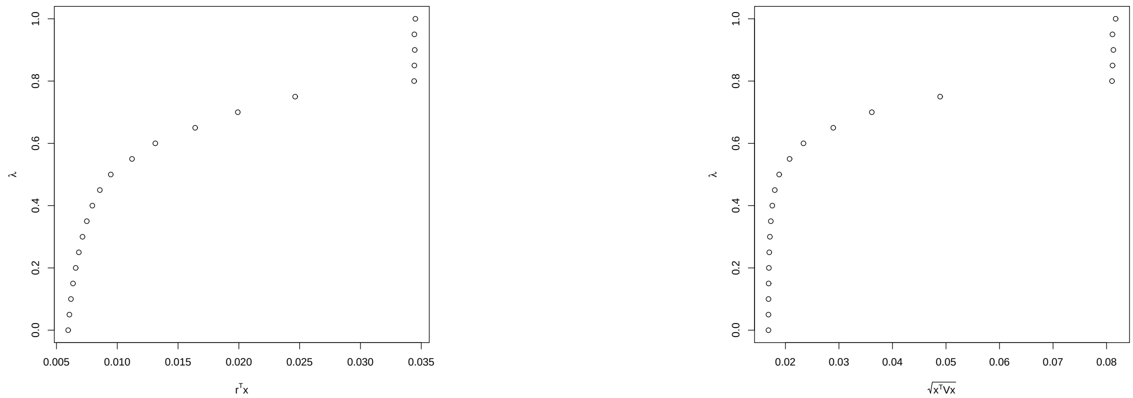
Obrázek 4.7: Efektivní hranice portfolia (21 a 101 pozorování, $konst_2 = 0{,}15$).



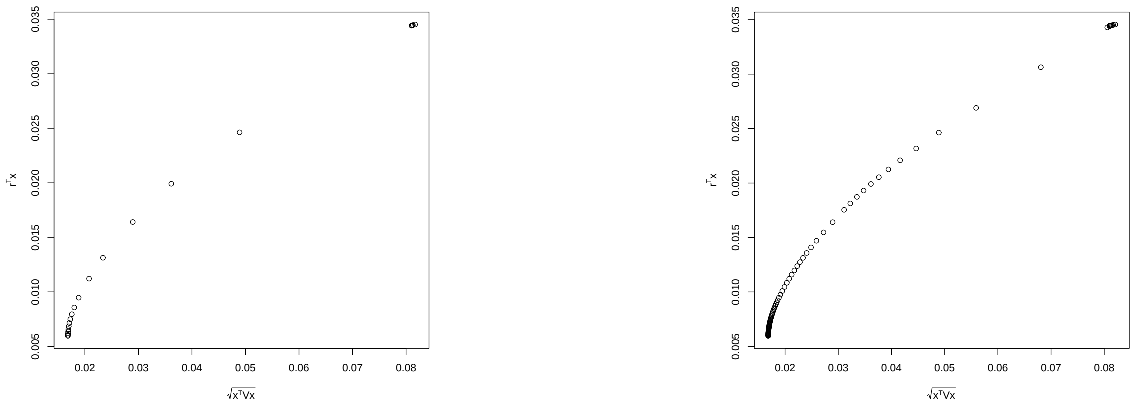
Obrázek 4.8: Efektivní hranice portfolia (101 poz., $konst_2 = 0{,}1$ a $konst_2 = 0{,}2$).

4.2.2 Úloha (4.2) - riziko $\sqrt{x^T V x}$

A) $X = \{x \in R^{24} \mid \sum_{i=1}^{24} x_i = 1,\ x_i \geq 0\ \forall i\}$



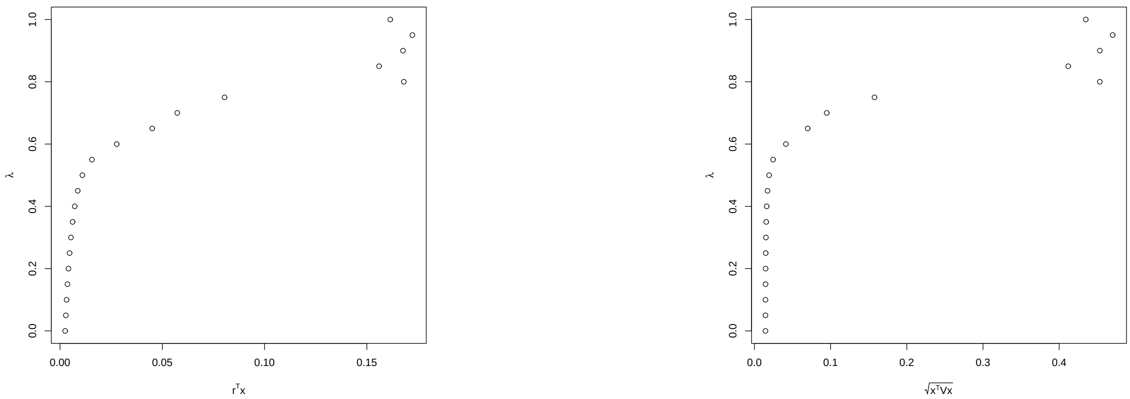
Obrázek 4.9: Závislost očekávaného výnosu a směrodatné odchylky výnosnosti na λ.



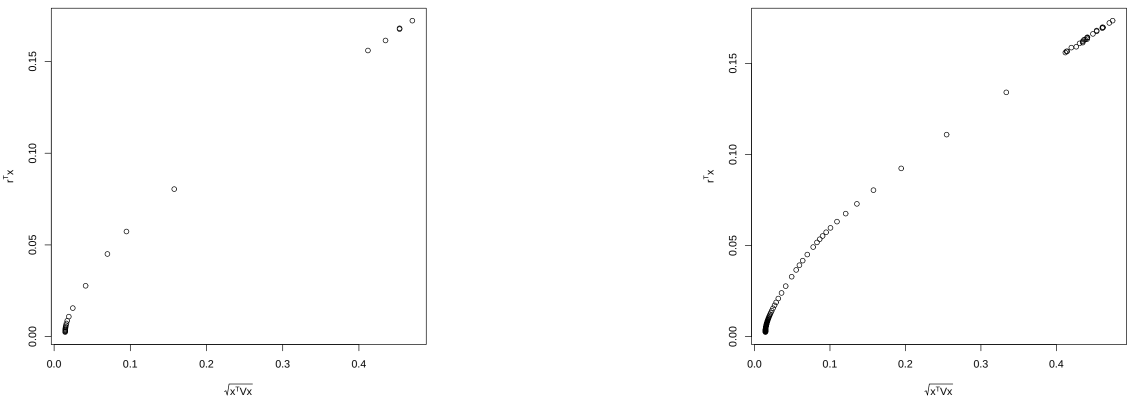
Obrázek 4.10: Efektivní hranice portfolia (21 a 101 pozorování).

B) $X = \{x \in R^{24} \mid \sum_{i=1}^{24} x_i = 1,\ x_i \geq -konst_1\ \forall i\}$

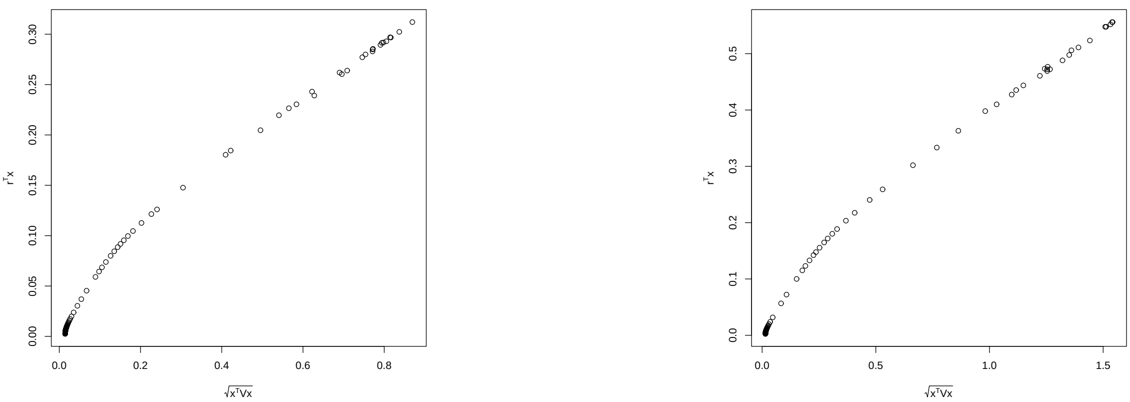
Opět si uvedeme grafické výstupy pro $konst_1 = 0{,}25$. Jsou tedy povoleny krátké prodeje do výše 25%. Znázorníme si i závislost očekávané výnosnosti na rozptylu výnosnosti pro $konst_1 = 0{,}50$ a $konst_1 = 1$.



Obrázek 4.11: Závislost očekávaného výnosu a směrodatné odchylky výnosnosti na λ.



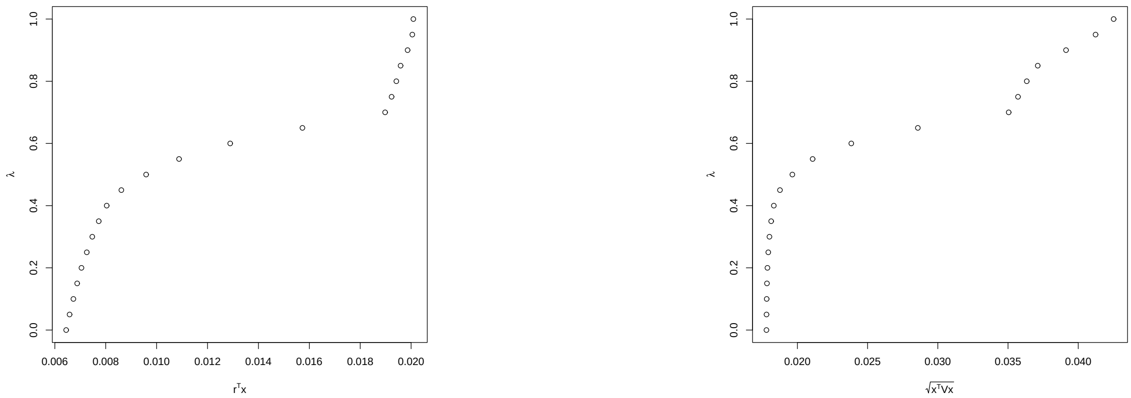
Obrázek 4.12: Efektivní hranice portfolia (21 a 101 pozorování, $konst_1 = 0{,}25$).



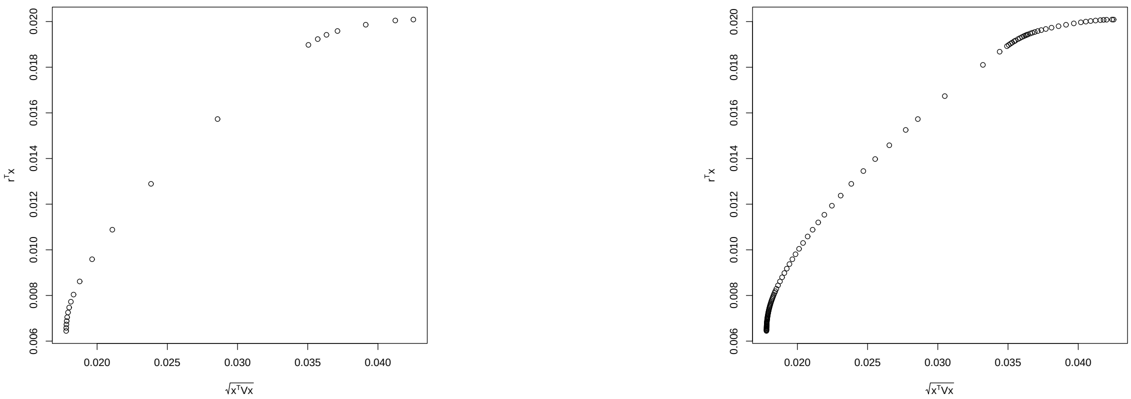
Obrázek 4.13: Efektivní hranice portfolia (101 pozorování, $konst_1 = 0{,}5$ a $konst_1 = 1$).

C) $X = \{x \in R^{24} \mid \sum_{i=1}^{24} x_i = 1,\ x_i \geq 0\ \forall i,\ x_i \leq konst_2\ \forall i\}$

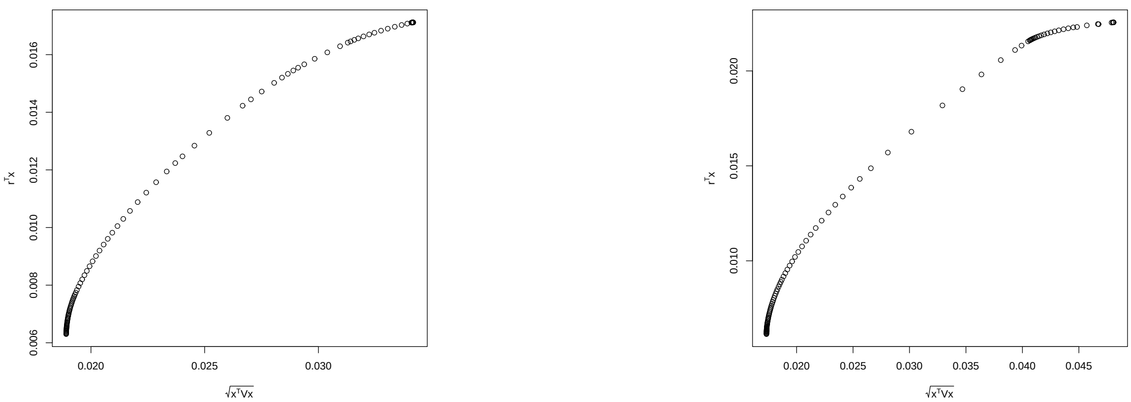
Opět si představíme grafické výstupy pro $konst_2 = 0{,}15$. Do žádného titulu tedy nesmíme investovat více než 15% z našeho majetku. Předvedeme si i efektivní hranici portfolia pro $konst_2 = 0{,}10$ a $konst_2 = 0{,}20$.



Obrázek 4.14: Závislost očekávaného výnosu a směrodatné odchylky výnosnosti na λ.



Obrázek 4.15: Efektivní hranice portfolia (21 a 101 pozorování, $konst_2 = 0{,}15$).



Obrázek 4.16: Efektivní hranice portfolia (101 poz., $konst_2 = 0{,}1$ a $konst_2 = 0{,}2$).

Poznámka: Předpokládáme, že uvedené $konst_1$ a $konst_2$ jsou kladné konstanty. Aby měly v našem příkladu smysl, volíme $konst_1$ libovolně a $konst_2 \in (\frac{1}{24}, 1)$.

Poznámka: Pro ověření myšlenky o nejméně riskantním řešení x_M (viz strana 52) jsme vypočítali i toto zmiňované x_M. Skutečně i v našem případě platí, že žádné jiné řešení z úlohy 4.2.1 neposkytuje menší očekávaný výnos.

Všechny výpočty byly provedeny pomocí volně přístupného softwaru R, konkrétně verzí R 2.2.0.

Kapitola 5

Závěr

Tato diplomová práce se zabývá úlohami vícekriteriální optimalizace. Jak napovídá název, rozhodnutí činíme pomocí více protichůdných kriterií. Jsou zde rozebrány jak úlohy deterministické (nezávisí na náhodě), tak i stochastické (na náhodě závisí). V úvodní kapitole jsme si ukázali motivaci a velmi stručně si představili, čím se zabývá vědní obor optimalizace. Je zde také vysvětleno značení, které v této práci používáme.

Druhá kapitola se věnuje vícekriteriální deterministické optimalizaci, která je nutným základem pro vícekriteriální stochastické úlohy. Po dvou příkladech a stručném přehledu vícekriteriálního hodnocení variant převážně rozebíráme vícekriteriální programování. Nejdříve popisujeme vztahy mezi uvedenými pojmy (eficientní řešení, slabě eficientní řešení, vlastní eficientní řešení a jejich funkční hodnoty – nedominovaná řešení), následně se seznámíme s alternativními definicemi vlastních eficientních řešení. Na závěr si ukážeme postupy hledání eficientních řešení.

V kapitole 3 se již objevují náhodné elementy. Představujeme různé situace, se kterými se můžeme ve vícekriteriální stochastické optimalizaci setkat. Ukazujeme si různé metody, jak převést stochastickou úlohu na deterministickou. Zmiňujeme se o modelech s pravděpodobnostními omezeními a s penalizací ztrát. Na závěr této části jsou i konkrétní příklady.

Aplikace vysvětlovaných postupů je v kapitole 4. Jde o příklad vícekriteriální optimalizační úlohy s náhodnými elementy, kde pracujeme s reálnými daty. Úkolem bylo sestavit optimální portfolio akcií, když máme k dispozici informace o těchto akciích z minulosti. Budoucí výnosnost však samozřejmě neznáme. Ta představuje onu náhodu. Chceme z naší investice co největší zisk při co nejmenším riziku. Graficky jsme znázornili efektivní hranice portfolií (závislost očekávaného zisku na riziku) pro různě obměňovaná omezení.

Tato diplomová práce přináší poměrně ucelený pohled na vícekriteriální optimalizaci. Shrnuje přehled přístupů, pomocí nichž lze získat řešení úloh jak deterministických, tak i stochastických. Práci by bylo možné rozšířit o vyšetřování vlastností řešení vícekriteriálních stochastických úloh, například stability množin eficientních řešení.

Literatura

[1] Arrow K. J., Barankin E. W., Blackwell D. (1953): *Admissible points of convex sets.* In "Contributions to the Theory of Games" (H. W. Kuhn and A. W. Tucker, eds.). Princeton Univ. Press, Princeton, New Jersey, 87–91.

[2] Benson H. P. (2001): *Multi-Objective Optimization: Pareto Optimal Solutions, Properties.* Encyclopedia of Optimization, Volume III, Kluwer, Dodrecht.

[3] Caballero R., Cerdá E., Muñoz M. M., Rey L., Stancu-Minasian I. M. (2001): *Efficient Solution Concepts and Their Relations in Stochastic Multiobjective Programming.* Journal of Optimization Theory and Applications 110, 1, 53–74.

[4] Dupačová J. (1986): *Stochastické programování.* Ministerstvo školství ČSR, Praha.

[5] Dupačová J., Hurt J., Štěpán J. (2002): *Stochastic Modeling in Economics and Finance.* Kluwer, London.

[6] Ehrgott M. (2005): *Multicriteria Optimization.* Springer, Berlin.

[7] Fiala P., Jablonský J., Maňas M. (1997): *Vícekriteriální rozhodování.* VŠE v Praze.

[8] Geoffrion A. M. (1968): *Proper Efficiency and the Theory of Vector Maximization.* Journal of Mathematical Analysis and Applications 22, 618–630.

[9] Grygarová L. (1996): *Základy vícekriteriálního programování.* UK - vydavatelství Karolinum, Praha.

[10] Guddat J., Vasquez F. G., Tammer K., Wendler K. (1985): *Multiobjective and Stochastic Optimization. Based on Parametric Optimization.* Akademie Verlag, Berlin.

[11] Cho G.-M. (1995): *Stability of the multiple objective linear stochastic programming problems.* Bull. Korean Math. Soc. 32, 2, 287–296.

[12] Isermann H. (1974): *Proper Efficiency and the Linear Vector Maximum Problem.* Operations Research 22, 1, 189–191.

[13] Kaňková V. (2004): *Multiobjective Programs and Markowitz Model.* In "Quantitative Methods in Economics. Proceedings of the International Conference" (M. Lukáčik, eds.). The Slovak Society for Operations Research, Bratislava, 109–117.

[14] Kaňková V. (2000): *Stochastic Programming Approach to Multiobjective Optimization Problems with Random Elements I.* Research Report UTIA AS CR, No. 1990.

[15] Kataoka S. (1963): *A Stochastic Programming Model.* Econometrica 31, 181–196.

[16] Stancu-Minasian I. M. (1984): *Stochastic Programming with Multiple Objective Function.* D. Reidel Publishing Company, Dodrecht.

[17] Takayama A. (1985): *Mathematical economics.* Cambridge University Press, Cambridge.

[18] Vyvialová S. (2002): *Vícekriteriální optimalizace a stochastické programování.* Rigorózní práce MFF UK v Praze.

Příloha

V příloze si uvedeme konkrétní hodnoty odhadů r a V z numerické ilustrace 4.2. Dále si na ukázku vypíšeme kompletní výsledky úlohy 4.2.1 A) a její zdrojový kód v programu R.

Kompletní výsledky a zdrojové kódy všech předváděných úloh jsou na přiloženém CD (1Areseni, 1Aprogram – 2Creseni, 2Cprogram). Na tomto disku je i výpočet nejméně riskantního řešení (nejmriziko) a data (data), z nichž jsou získané odhady r (rtabulka) a V (Vtabulka).

Odhady r a V

Konkrétní hodnoty odhadů středních výnosů r a rozptylů výnosnosti V u numerické ilustrace 4.2:

$$
\begin{array}{lllllll}
r = (& 0,004294214; & 0,002341778; & 0,014432819; & 0,035515110; & 0,003670470; & 0,010988477 \\
 & 0,015546206; & 0,001992241; & 0,009042404; & 0,007744289; & 0,033102064; & 0,009921207 \\
 & 0,014132388; & 0,005274853; & 0,008139782; & 0,004534941; & 0,011982324; & 0,001992360 \\
 & 0,012258259; & 0,009500920; & 0,007938718; & 0,003882376; & 0,013003936; & 0,010181844\ )
\end{array}
$$

Matice V je tvořena čtyřmi podmaticemi: $V = (V_1\,|\,V_2|\,V_3|\,V_4)$.

$$
V_1 = \begin{pmatrix}
0,000582662 & -0,000023055 & 0,000241724 & -0,000066233 & 0,000205666 & 0,000236225 \\
-0,000023055 & 0,001795058 & 0,000372473 & 0,000595002 & 0,000123943 & 0,000237933 \\
0,000241724 & 0,000372473 & 0,005812027 & 0,002940528 & 0,000714985 & 0,000213627 \\
-0,000066233 & 0,000595002 & 0,002940528 & 0,012905231 & 0,000712749 & 0,000055448 \\
0,000205666 & 0,000123943 & 0,000714985 & 0,000712749 & 0,000859492 & 0,000201326 \\
0,000236225 & 0,000237933 & 0,000213627 & 0,000055448 & 0,000201326 & 0,004517270 \\
0,000005416 & 0,000217953 & 0,000256728 & 0,001065256 & 0,000160050 & -0,000125042 \\
0,000184254 & 0,000601825 & 0,000587948 & 0,000136811 & 0,000201252 & 0,001066290 \\
0,000384699 & 0,000291539 & 0,000518904 & 0,000050585 & 0,000509123 & 0,000432979 \\
0,000295511 & 0,001154558 & 0,000609531 & 0,001059945 & 0,000142618 & 0,001002751 \\
0,000773109 & 0,000872604 & 0,000903582 & 0,000167173 & 0,000291196 & 0,000366921 \\
0,000380008 & 0,000436587 & 0,000704572 & -0,000129616 & 0,000425259 & 0,000690810 \\
0,000328634 & 0,000772985 & 0,001255649 & 0,000147258 & 0,000615635 & 0,000855350 \\
0,000154648 & 0,000326451 & 0,000390783 & 0,000224880 & 0,000230160 & 0,000433694 \\
0,000086742 & 0,000254639 & 0,000072593 & 0,000584696 & 0,000035432 & -0,000185955 \\
0,000320358 & 0,000334544 & 0,000554828 & 0,000614552 & 0,000221929 & 0,001049919 \\
0,000167122 & 0,000375480 & 0,000544965 & 0,000804842 & 0,000321189 & -0,000062417 \\
0,000370072 & 0,000240029 & 0,000717380 & 0,000325262 & 0,000303329 & 0,000901039 \\
0,000574389 & 0,000601722 & 0,000394486 & 0,000084903 & 0,000724287 & 0,001134316 \\
0,000384217 & 0,000498325 & 0,000656668 & 0,000428503 & 0,000413930 & 0,000796677 \\
0,000253589 & 0,000573848 & 0,000586563 & 0,000847133 & 0,000478038 & 0,000622079 \\
0,000279277 & 0,000166275 & 0,000707791 & 0,000506981 & 0,000352613 & 0,000362949 \\
0,000345746 & 0,000859940 & 0,000826394 & 0,000845157 & 0,000387087 & 0,001053348 \\
0,000053835 & 0,000390049 & 0,000127703 & 0,000161341 & 0,000160714 & 0,000494880
\end{pmatrix}
$$

$$V_2 = \begin{pmatrix}
0,000005416 & 0,000184254 & 0,000384699 & 0,000295511 & 0,000773109 & 0,000380008 \\
0,000217953 & 0,000601825 & 0,000291539 & 0,001154558 & 0,000872604 & 0,000436587 \\
0,000256728 & 0,000587948 & 0,000518904 & 0,000609531 & 0,000903582 & 0,000704572 \\
0,001065256 & 0,000136811 & 0,000050585 & 0,001059945 & 0,000167173 & -0,000129616 \\
0,000160050 & 0,000201252 & 0,000509123 & 0,000142618 & 0,000291196 & 0,000425259 \\
-0,000125042 & 0,001066290 & 0,000432979 & 0,001002751 & 0,000366921 & 0,000690810 \\
0,001709235 & 0,000226780 & -0,000049336 & 0,000910153 & 0,001440849 & 0,000523054 \\
0,000226780 & 0,001567047 & 0,000182853 & 0,000513637 & 0,000877693 & 0,000505843 \\
-0,000049336 & 0,000182853 & 0,002285586 & 0,000539311 & 0,000773949 & 0,001112291 \\
0,000910153 & 0,000513637 & 0,000539311 & 0,004716849 & 0,001672376 & 0,001018830 \\
0,001440849 & 0,000877693 & 0,000773949 & 0,001672376 & 0,012866456 & 0,000858232 \\
0,000523054 & 0,000505843 & 0,001112291 & 0,001018830 & 0,000858232 & 0,003824273 \\
0,000344700 & 0,000834598 & 0,001217273 & 0,001202468 & 0,001933885 & 0,000896229 \\
0,000384523 & 0,000492721 & 0,000140446 & 0,000520651 & 0,000985746 & 0,000491847 \\
0,000223392 & 0,000139532 & 0,000327261 & 0,000042339 & 0,000699424 & 0,000278941 \\
0,000446688 & 0,000872456 & 0,000510953 & 0,001285039 & 0,002337174 & 0,000732407 \\
-0,000105580 & -0,000012424 & 0,000940646 & 0,000254178 & -0,000484122 & 0,000813941 \\
0,000186680 & 0,000546692 & 0,000619377 & 0,000946394 & 0,001053001 & 0,000685058 \\
0,000364388 & 0,000732808 & 0,000959200 & 0,001054121 & 0,001699437 & 0,001020285 \\
-0,000127215 & 0,000680734 & 0,000211492 & 0,001153495 & 0,001598774 & 0,000441310 \\
0,000363145 & 0,000388455 & 0,000460030 & 0,001204305 & 0,001873812 & 0,000266654 \\
0,000109995 & 0,000433375 & 0,000349196 & 0,000278257 & 0,000719004 & 0,000554206 \\
0,000428555 & 0,001117444 & 0,000890445 & 0,001483291 & 0,002590878 & 0,000890328 \\
0,000181661 & 0,000405643 & 0,000669525 & 0,000051778 & -0,000230768 & 0,001008429
\end{pmatrix}$$

$$V_3 = \begin{pmatrix}
0,000328634 & 0,000154648 & 0,000086742 & 0,000320358 & 0,000167122 & 0,000370072 \\
0,000772985 & 0,000326451 & 0,000254639 & 0,000334544 & 0,000375480 & 0,000240029 \\
0,001255649 & 0,000390783 & 0,000072593 & 0,000554828 & 0,000544965 & 0,000717380 \\
0,000147258 & 0,000224880 & 0,000584696 & 0,000614552 & 0,000804842 & 0,000325262 \\
0,000615635 & 0,000230160 & 0,000035432 & 0,000221929 & 0,000321189 & 0,000303329 \\
0,000855350 & 0,000433694 & -0,000185955 & 0,001049919 & -0,000062417 & 0,000901039 \\
0,000344700 & 0,000384523 & 0,000223392 & 0,000446688 & -0,000105580 & 0,000186680 \\
0,000834598 & 0,000492721 & 0,000139532 & 0,000872456 & -0,000012424 & 0,000546692 \\
0,001217273 & 0,000140446 & 0,000327261 & 0,000510953 & 0,000940646 & 0,000619377 \\
0,001202468 & 0,000520651 & 0,000042339 & 0,001285039 & 0,000254178 & 0,000946394 \\
0,001933885 & 0,000985746 & 0,000699424 & 0,002337174 & -0,000484122 & 0,001053001 \\
0,000896229 & 0,000491847 & 0,000278941 & 0,000732407 & 0,000813941 & 0,000685058 \\
0,003564608 & 0,000732617 & 0,000158321 & 0,001124442 & 0,000323089 & 0,000974743 \\
0,000732617 & 0,000827510 & 0,000040005 & 0,000383070 & 0,000146312 & 0,000320486 \\
0,000158321 & 0,000040005 & 0,001999570 & -0,000478910 & 0,000755052 & 0,000009467 \\
0,001124442 & 0,000383070 & -0,000478910 & 0,003687214 & -0,000129250 & 0,000875227 \\
0,000323089 & 0,000146312 & 0,000755052 & -0,000129250 & 0,004053489 & 0,000371614 \\
0,000974743 & 0,000320486 & 0,000009467 & 0,000875227 & 0,000371614 & 0,001464979 \\
0,001467118 & 0,000514001 & 0,000377113 & 0,001470061 & 0,000004550 & 0,001074839 \\
0,001299802 & 0,000524405 & 0,000728545 & 0,000669835 & -0,000174404 & 0,000841503 \\
0,001013387 & 0,000301201 & 0,000188391 & 0,001239364 & -0,000108544 & 0,000693633 \\
0,000596171 & 0,000289330 & 0,000346167 & 0,000448688 & 0,000501346 & 0,000617914 \\
0,001423792 & 0,000606097 & 0,000306184 & 0,001570977 & 0,000559581 & 0,000850872 \\
0,000482862 & 0,000069217 & 0,000318258 & 0,000753554 & 0,000717702 & 0,000637499
\end{pmatrix}$$

$$V_4 = \begin{pmatrix}
0,000574389 & 0,000384217 & 0,000253589 & 0,000279277 & 0,000345746 & 0,000053835 \\
0,000601722 & 0,000498325 & 0,000573848 & 0,000166275 & 0,000859940 & 0,000390049 \\
0,000394486 & 0,000656668 & 0,000586563 & 0,000707791 & 0,000826394 & 0,000127703 \\
0,000084903 & 0,000428503 & 0,000847133 & 0,000506981 & 0,000845157 & 0,000161341 \\
0,000724287 & 0,000413930 & 0,000478038 & 0,000352613 & 0,000387087 & 0,000160714 \\
0,001134316 & 0,000796677 & 0,000622079 & 0,000362949 & 0,001053348 & 0,000494880 \\
0,000364388 & -0,000127215 & 0,000363145 & 0,000109995 & 0,000428555 & 0,000181661 \\
0,000732808 & 0,000680734 & 0,000388455 & 0,000433375 & 0,001117444 & 0,000405643 \\
0,000959200 & 0,000211492 & 0,000460030 & 0,000349196 & 0,000890445 & 0,000669525 \\
0,001054121 & 0,001153495 & 0,001204305 & 0,000278257 & 0,001483291 & 0,000051778 \\
0,001699437 & 0,001598774 & 0,001873812 & 0,000719004 & 0,002590878 & -0,000230768 \\
0,001020285 & 0,000441310 & 0,000266654 & 0,000554206 & 0,000890328 & 0,001008429 \\
0,001467118 & 0,001299802 & 0,001013387 & 0,000596171 & 0,001423792 & 0,000482862 \\
0,000514001 & 0,000524405 & 0,000301201 & 0,000289330 & 0,000606097 & 0,000069217 \\
0,000377113 & 0,000728545 & 0,000188391 & 0,000346167 & 0,000306184 & 0,000318258 \\
0,001470061 & 0,000669835 & 0,001239364 & 0,000448688 & 0,001570977 & 0,000753554 \\
0,000004550 & -0,000174404 & -0,000108544 & 0,000501346 & 0,000559581 & 0,000717702 \\
0,001074839 & 0,000841503 & 0,000693633 & 0,000617914 & 0,000850872 & 0,000637499 \\
0,003905000 & 0,001342098 & 0,001404153 & 0,000885107 & 0,001251722 & 0,000538474 \\
0,001342098 & 0,008218993 & 0,001190506 & 0,000950673 & 0,001144862 & -0,000355249 \\
0,001404153 & 0,001190506 & 0,002796025 & 0,000372937 & 0,000887226 & 0,000687039 \\
0,000885107 & 0,000950673 & 0,000372937 & 0,001162277 & 0,000708079 & 0,000263988 \\
0,001251722 & 0,001144862 & 0,000887226 & 0,000708079 & 0,003020470 & 0,000417913 \\
0,000538474 & -0,000355249 & 0,000687039 & 0,000263988 & 0,000417913 & 0,004377027
\end{pmatrix}$$

Výsledky

Řešení, očekávaná výnosnost a rozptyl výnosnosti pro 21 hodnot

$$\lambda = \{0{,}00;\ 0{,}05;\ 0{,}10;\ \ldots;\ 0{,}95;\ 1{,}00\}$$

úlohy 4.2.1 A):

```
NUMERICKÉ VÝSLEDKY ÚLOHY 4.2.1 A

> RESENI  # reseni pro vsechny hodnoty parametru lambda
              [,1]         [,2]         [,3]        [,4]         [,5]         [,6]       [,7]         [,8]
 [1,] 1.892511e-01 6.520667e-02 1.121285e-02 0.008397298 1.260340e-01 2.248989e-02 0.10902264 3.452989e-02
 [2,] 1.284866e-01 1.310446e-02 2.001942e-02 0.044042054 6.009384e-02 5.824689e-02 0.18582309 8.668757e-03
 [3,] 4.595427e-02 1.484040e-07 1.218250e-02 0.106432084 3.220946e-06 9.886029e-02 0.32733310 3.411262e-08
 [4,] 7.844831e-03 3.266769e-03 1.293431e-02 0.165980186 4.836830e-03 7.717258e-02 0.27367416 3.175571e-03
 [5,] 4.002639e-03 3.816164e-03 8.126884e-03 0.249370073 2.889344e-03 6.319210e-02 0.16546528 2.230619e-03
 [6,] 1.159531e-04 4.698890e-06 1.136264e-03 0.326217404 8.249291e-05 4.518779e-02 0.15080912 2.700315e-05
 [7,] 1.441288e-03 1.720938e-03 4.732498e-02 0.291760678 1.310560e-03 3.916673e-02 0.10797839 1.171796e-03
 [8,] 1.313577e-03 9.203044e-04 5.843440e-04 0.424227898 1.115119e-03 5.259804e-03 0.10267395 9.319472e-04
 [9,] 1.069079e-03 4.460253e-04 6.511136e-03 0.430427411 9.395170e-04 3.066718e-03 0.09732261 8.241201e-04
[10,] 8.880763e-04 5.101307e-04 1.229423e-02 0.436562752 7.263354e-04 1.897857e-03 0.08630101 5.569254e-04
[11,] 5.991858e-06 2.242962e-06 2.831376e-06 0.503071077 1.932421e-06 1.220527e-05 0.05586281 2.166210e-06
[12,] 3.647728e-04 9.584612e-04 5.997498e-02 0.234714714 6.956404e-04 4.364238e-02 0.08308597 7.995379e-04
[13,] 6.334609e-04 5.519399e-04 4.172091e-02 0.391277291 5.895980e-04 7.847174e-03 0.08312427 3.597388e-04
[14,] 6.945173e-04 4.287267e-04 6.245630e-02 0.264003446 6.686154e-04 3.779334e-02 0.08270290 5.090133e-04
[15,] 3.895037e-04 8.177269e-05 3.895235e-02 0.415098439 3.772922e-04 1.903475e-03 0.07499148 1.453517e-04
[16,] 1.781211e-04 2.279873e-04 2.625025e-02 0.462010264 2.003683e-04 6.279707e-04 0.06192887 2.898379e-04
[17,] 1.829472e-04 2.597264e-04 1.124721e-02 0.492247635 2.946160e-04 2.491496e-04 0.04605055 2.486833e-04
[18,] 2.509407e-04 2.781380e-04 4.585808e-02 0.414352934 3.034253e-04 5.063869e-04 0.07133919 2.419722e-04
[19,] 2.261769e-04 2.470214e-04 4.867300e-02 0.411658105 2.821862e-04 3.187899e-04 0.07119024 2.537441e-04
[20,] 1.680316e-04 3.698600e-04 4.344900e-02 0.432521866 3.583201e-04 1.373997e-04 0.06561543 1.562230e-04
[21,] 1.984146e-04 2.498783e-04 2.563985e-02 0.476829654 1.073666e-04 1.407295e-04 0.04920080 2.256791e-04
```

```
             [,9]        [,10]       [,11]        [,12]        [,13]        [,14]        [,15]        [,16]
 [1,] 1.803329e-02 6.285609e-03 0.005018378 7.788230e-03 0.0087098434 1.131334e-01 1.026529e-01 1.845106e-02
 [2,] 3.765435e-02 7.175193e-03 0.024637625 1.073206e-02 0.0163334085 7.759467e-02 9.866106e-02 8.064232e-03
 [3,] 4.195542e-02 5.193449e-09 0.084689418 2.602529e-05 0.0451014671 1.171396e-05 3.472506e-02 5.738796e-10
 [4,] 1.782605e-02 3.371626e-03 0.140805419 1.033880e-02 0.0423558670 5.714074e-03 1.376925e-02 2.653753e-03
 [5,] 9.823307e-03 2.577076e-03 0.217792189 1.029896e-02 0.0537485667 3.471520e-03 7.354103e-03 1.901084e-03
 [6,] 2.400999e-03 1.205779e-04 0.285536445 6.879448e-03 0.0507508309 1.184593e-04 1.674355e-04 3.687667e-05
 [7,] 1.072005e-02 1.398746e-03 0.249436127 1.482920e-02 0.0564858344 1.447978e-03 9.092243e-03 7.012543e-04
 [8,] 2.139767e-03 1.251723e-03 0.372409324 2.465338e-03 0.0327442683 1.345410e-03 1.742912e-03 1.136395e-03
 [9,] 1.650249e-03 3.287751e-04 0.378759260 1.523096e-03 0.0386717632 1.146169e-03 1.200975e-03 2.901543e-04
[10,] 1.248074e-03 8.695233e-04 0.384719016 1.465968e-03 0.0365894977 9.220498e-04 3.764372e-03 6.915482e-04
[11,] 6.212563e-06 2.048417e-05 0.440375413 6.206324e-06 0.0005666196 5.749122e-06 3.733830e-06 1.500036e-06
[12,] 2.378656e-02 4.526590e-03 0.213192826 3.076518e-02 0.0645849500 4.991505e-04 1.665057e-02 7.009653e-04
[13,] 4.187453e-04 4.893041e-04 0.349708552 7.183717e-04 0.0504484016 5.719525e-04 7.560171e-04 4.555455e-04
[14,] 1.681684e-02 3.413061e-04 0.239263427 2.480574e-02 0.0641147118 5.829487e-04 8.915767e-03 1.039355e-04
[15,] 1.009227e-03 5.410190e-04 0.371834428 9.635363e-04 0.0446681417 4.406470e-04 7.599830e-04 2.959064e-04
[16,] 4.947326e-04 3.842221e-04 0.411180768 5.965951e-04 0.0285317546 2.714420e-04 5.533451e-04 2.699458e-04
[17,] 2.904565e-05 1.422934e-04 0.435716355 2.024892e-04 0.0104602646 3.658381e-04 3.514877e-04 1.628431e-04
[18,] 3.423790e-04 4.134772e-04 0.371969850 1.444132e-04 0.0449733923 3.481432e-04 2.622938e-04 3.154940e-04
[19,] 2.451835e-04 1.961895e-04 0.369584431 1.022130e-04 0.0455475577 2.290078e-04 2.224269e-04 1.903951e-04
[20,] 2.727810e-04 2.496315e-04 0.387714609 3.474801e-05 0.0390280870 6.195195e-04 3.451420e-04 3.665863e-04
[21,] 9.583859e-05 5.811498e-05 0.425040141 1.085616e-04 0.0193434789 2.108292e-05 4.153365e-04 1.017194e-04
            [,17]        [,18]        [,19]        [,20]        [,21]        [,22]        [,23]        [,24]
 [1,] 1.982695e-02 1.755186e-02 4.516933e-03 8.833243e-03 1.457104e-02 5.416910e-02 1.045776e-02 2.385598e-02
 [2,] 6.159447e-02 1.935325e-02 1.363621e-02 1.701815e-02 1.183684e-02 1.837095e-02 1.309352e-02 4.575877e-02
 [3,] 1.162075e-01 2.825780e-08 7.937728e-03 1.992195e-02 8.781078e-07 6.617995e-07 4.099854e-06 5.865251e-02
 [4,] 1.140896e-01 2.949539e-03 1.991108e-02 1.261775e-02 4.942788e-03 4.364032e-03 7.032892e-03 4.837231e-02
 [5,] 9.566067e-02 2.037897e-03 2.260210e-02 6.619725e-03 2.928223e-03 2.742679e-03 6.655609e-03 5.469326e-02
 [6,] 8.066566e-02 4.142154e-05 1.292370e-02 3.771176e-04 6.932136e-05 6.857053e-05 3.144029e-04 3.594811e-02
 [7,] 6.219044e-02 1.062817e-03 3.305002e-02 9.148594e-04 1.396159e-03 1.336535e-03 2.848971e-02 3.557275e-02
 [8,] 3.539242e-02 9.873214e-04 2.134332e-03 1.870503e-03 1.768075e-03 1.098838e-03 1.873424e-03 2.613098e-03
 [9,] 3.043169e-02 8.869056e-04 9.557052e-04 1.329764e-03 1.863913e-04 8.007261e-04 1.111407e-03 1.204512e-04
[10,] 2.190951e-02 7.422513e-04 1.921647e-03 1.450023e-03 1.105046e-03 6.286485e-04 6.474219e-04 1.588165e-03
[11,] 5.188978e-06 8.344651e-06 9.450844e-06 1.404661e-05 5.879409e-06 3.475495e-06 1.588231e-06 4.946647e-06
[12,] 5.394018e-02 4.264556e-04 4.876387e-02 2.255881e-02 7.621164e-03 6.899040e-04 4.956859e-02 3.748787e-02
[13,] 2.688815e-02 4.480558e-04 1.895490e-02 6.188869e-04 6.331318e-04 5.820589e-04 2.109841e-02 1.105234e-03
[14,] 4.889641e-02 4.765361e-04 4.599393e-02 1.724135e-02 2.337246e-03 8.134728e-04 4.963282e-02 3.040680e-02
[15,] 1.532953e-02 1.575099e-04 1.189705e-02 6.325581e-04 6.287651e-04 2.943191e-04 1.681705e-02 1.790761e-03
[16,] 1.053072e-03 3.301915e-04 8.976477e-04 5.266753e-04 2.733123e-04 3.381245e-04 1.743352e-03 8.412478e-04
[17,] 2.646417e-05 2.252004e-04 5.464766e-05 2.441034e-04 2.487255e-04 2.872691e-04 1.573364e-04 5.452015e-04
[18,] 1.152716e-02 2.614103e-04 1.273220e-02 3.098050e-04 3.514714e-04 2.612574e-04 2.227956e-02 3.767249e-04
[19,] 1.123931e-02 2.108844e-04 1.376248e-02 3.008232e-05 1.997932e-04 2.970844e-04 2.476629e-02 3.275025e-04
[20,] 3.257375e-03 3.380493e-05 6.001462e-03 3.402706e-04 4.315463e-04 4.181736e-04 1.781614e-02 2.940873e-04
[21,] 2.575613e-04 2.260062e-04 4.929594e-04 1.138543e-04 4.332265e-04 2.677152e-04 2.915167e-04 1.406135e-04

> VYNOS  # ocekavana vynosnost pro vsechny hodnoty parametru lambda
 [1] 0.007109797 0.010907044 0.016707644 0.019330220 0.022715913 0.026205667 0.024476273 0.030152134 0.030483053
[10] 0.030713742 0.033321443 0.022196027 0.029052977 0.023580319 0.030046128 0.031858166 0.032960462 0.030052302
[19] 0.029954456 0.030766769 0.032443211

> ROZPTYL  # rozptyl vynosnosti pro vsechny hodnoty parametru lambda
 [1] 0.0003424932 0.0004680689 0.0008707370 0.0012666960 0.0019821484 0.0029128714 0.0025233122 0.0044938109
 [9] 0.0046286589 0.0047533971 0.0059726846 0.0020166456 0.0041034619 0.0023469137 0.0045114041 0.0052997178
[17] 0.0058189835 0.0045353633 0.0044992113 0.0048455642 0.0055802994
```

Zdrojový kód

Zdrojový kód programu R pro příklad 4.2.1 A):

```
ZDROJOVÝ KÓD ÚLOHY 4.2.1 A

> v1<-cbind(0.000582662,-0.000023055, ... (#nacteni vsech hodnot) ,0.000345746,0.000053835)
> v2<-cbind(-0.000023055,0.001795058, ... (#nacteni vsech hodnot) ,0.00085994,0.000390049)

  ... (#nacteni vsech hodnot)

> v24<-cbind(0.000053835,0.000390049, ... (#nacteni vsech hodnot) ,0.000417913,0.004377027)
> V<-rbind(v1,v2,v3,v4,v5,v6,v7,v8,v9,v10,v11,v12,v13,v14,v15,v16,v17,v18,v19,v20,v21,v22,v23,v24)
> V  # matice rozptylu ocekavane vynosnosti
> r<-cbind(0.004294214,0.002341778, ... (#nacteni vsech hodnot) ,0.013003936,0.010181844)
> r  # odhad strednich vynosu
> tr<-rbind(0.004294214,0.002341778, ... (#nacteni vsech hodnot) ,0.013003936,0.010181844)
> tr  # transpozice r
> ui<-rbind(diag(24),rep(1,24),rep(-1,24))
> ci<-c(rep(0,24),0.999,-1.001)
> ui  # omezeni (leva strana)
> ci  # omezeni (prava strana)
> VYNOS<-1
> ROZPTYL<-1
> RESENI<-cbind(rep(1,21),1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)
> LAMBDA<-1
> for (n in seq(1,21,by=1)) {
+ LAMBDA[n]<-(n-1)/20
+ fr<-function(x) {
+ -LAMBDA[n]*r%*%x + (1-LAMBDA[n])*t(x)%*%V%*%x }
+ grr<-function(x) {
+ c(-LAMBDA[n]*tr + 2*(1-LAMBDA[n])*V%*%x) }
+ X<-constrOptim(c(0.0417,0.0417, ... (#celkem 24 "startovacich" hodnot reseni) ,0.0417,0.0409),fr,grr,ui,ci)
+ RESENI[n,]<-X[[1]]
+ VYNOS[n]<-X[[1]]%*%tr
+ ROZPTYL[n]<-X[[1]]%*%V%*%X[[1]]
+ }
> LAMBDA  # pouzite hodnoty parametru lambda
> VYNOS  # ocekavana vynosnost pro vsechny hodnoty parametru lambda
> ROZPTYL  # rozptyl vynosnosti pro vsechny hodnoty parametru lambda
> RESENI  # reseni pro vsechny hodnoty parametru lambda
> plot(LAMBDA~VYNOS)  # vykresli graf zavislosti ocek. vynosnosti na lambda
> plot(LAMBDA~ROZPTYL)  # graf zavislosti rozptylu vynosnosti na lambda
> plot(VYNOS~ROZPTYL)  # graf zavislosti ocek. vynosnosti na jejim rozptylu
```