

Univerzita Karlova v Praze  
Matematicko-fyzikální fakulta

## DIPLOMOVÁ PRÁCE



Peter Baník

### **Metody optimalizace ve financích**

Katedra pravděpodobnosti a matematické statistiky  
Vedoucí diplomové práce: Doc. RNDr. Jan Hurt, CSc.  
Studijní program: Finanční a pojistná matematika

2008

Chcel by som poďakovať vedúcemu diplomovej práce, Doc. RNDr. Janovi Hurtovi, CSc. za odborné vedenie, mnohé cenné rady a pripomienky k textu tejto diplomovej práce. Taktiež ďakujem všetkým svojim priateľom a rodine za technickú a morálnu podporu.

Prehlasujem, že som svoju diplomovú prácu napísal samostatne a výhradne s použitím citovaných prameňov. Súhlasím so zapožičaním práce.

V Prahe dňa 12.12.2008

Peter Baník

# Obsah

<b>1</b>	<b>Úvod</b>	<b>6</b>
1.1	Optimalizačný problém . . . . .	6
1.1.1	Lineárne programovanie (LP) . . . . .	7
1.1.2	Kvadratické programovanie (QP) . . . . .	8
1.2	Stochastické optimalizačné algoritmy . . . . .	8
1.3	Finančný management . . . . .	9
1.3.1	Problém optimálneho portfólia . . . . .	9
1.3.2	Ohraničenia v optimalizácii . . . . .	9
<b>2</b>	<b>Optimalizačné metódy všeobecne</b>	<b>11</b>
2.1	Lineárne programovanie . . . . .	11
2.1.1	Simplexový algoritmus . . . . .	11
2.2	Nelineárne programovanie . . . . .	12
2.2.1	Gradientné metódy . . . . .	12
2.3	Metódy vnútorného bodu (Interior-Point Methods) . . . . .	15
2.3.1	Metóda vnútorného bodu pre konvexné QP . . . . .	15
2.3.2	Metóda vnútorného bodu pre LP . . . . .	18
<b>3</b>	<b>Vybrané optimalizačné metódy</b>	<b>20</b>
3.1	Nelder-Meadova metóda (NM) . . . . .	20
3.1.1	Nelder-Meadov algoritmus . . . . .	21
3.2	Simulované tavenie (SA) . . . . .	22
3.2.1	Základný algoritmus . . . . .	23
3.3	Metóda diferenciálnych evolúcií (DE) . . . . .	25
3.3.1	Základný algoritmus . . . . .	26
<b>4</b>	<b>Aplikácie vo financiách a numerické ilustrácie</b>	<b>28</b>
4.1	Problém optimálneho portfólia a finančné umiestnenie aktív . . . . .	28
4.1.1	Základný model a reálne ohraničenia . . . . .	29
4.1.2	Alternatívne prístupy k problému optimálneho portfólia . . . . .	31
4.1.3	VaR a CVaR . . . . .	33
4.2	Numerické ilustrácie . . . . .	37
4.2.1	Numerické optimalizačné metódy . . . . .	37
4.2.2	Optimalizačné procedúry . . . . .	38
4.2.3	Ohraničenia v modeloch . . . . .	44
4.2.4	Výpočetné vlastnosti optimalizačných metód . . . . .	45

<b>5 Záver</b>	<b>49</b>
<b>Literatúra</b>	<b>50</b>

Názov práce: Metody optimalizace ve financích  
Autor: Peter Baník  
Katedra: Katedra pravděpodobnosti a matematické statistiky  
Vedúci diplomovej práce: Doc. RNDr. Jan Hurt, CSc.  
e-mail vedúceho: jan.hurt@mff.cuni.cz

Abstrakt: V tejto diplomovej práci sa venujeme vybraným metódam optimalizácie a modelom matematického programovania. Zameriavame sa na optimalizačné modely problému voľby optimálneho portfólia. K problému pristupujeme na základe kritérií maximálneho očakávaného výnosu a minimálneho rizika. Pre vybrané miery rizika používame vhodné matematické modely. Pojednávame o vhodných optimalizačných metódach pre lineárne a kvadratické programovanie. Pre všeobecne nelineárne problémy používame moderné stochastické optimalizačné algoritmy. Metódy numericky ilustrujeme na príkladoch s použitím reálnych dát, prostredníctvom softwarového systému Mathematica. Prezentované výsledky nám dávajú porovnania pre modely a tiež pre optimalizačné metódy. Diskutované sú výpočetné nároky optimalizačných algoritmov a vplyv vstupných parametrov na dosiahnuté výsledky.

Kľúčové slová: matematické programovanie, stochastické algoritmy, riziko, Mathematica

Title: Optimization methods in finance  
Author: Peter Baník  
Department: Department of Probability and Mathematical Statistics  
Supervisor: Doc. RNDr. Jan Hurt, CSc.  
Supervisor's e-mail address: jan.hurt@mff.cuni.cz

Abstract: In this diploma paper we discuss selected optimization methods and mathematical programming models. We focus on optimization models of optimal portfolio selection problem. We consider the problem of finding optimal portfolio under criterion of maximizing expected return and risk minimizing. For selected risk measures we use convenient mathematical models. There are adequate optimization techniques for problems of linear and quadratic programming. For general nonlinear problems we use advanced stochastic optimization algorithms. We numerically illustrate the methods on real data examples using the Mathematica software. Presented results give comparisons both for models and methods. We also focus on computational complexity of optimization algorithms and study the influence of input parameters on the results.

Keywords: mathematical programming, stochastic algorithms, risk, Mathematica

# Kapitola 1

## Úvod

Matematické programovanie je významnou súčasťou aplikovanej matematiky a finančného rozhodovania. Pomocou moderných optimalizačných metód a modelov dokážeme efektívne riešiť otázky optimálnych portfólií, finančného umiestnenia, riadenie finančných rizík a mnoho ďalších aplikácií. V posledných desaťročiach, s nástupom rozvoja výpočetnej techniky a ich výpočetných schopností, narastá význam a oblasť uplatnenia týchto techník, ktoré v súčasnej dobe dovoľujú riešiť problémy v rádoch miliónoch premenných.

V nasledujúcich kapitolách sa budeme venovať niektorým typom optimalizačných úloh v spojitosti s finančným rozhodovaním. Postupne prediskutujeme jednotlivé typy úloh, vhodné metódy k ich riešeniu a praktické úlohy finančnej matematiky riešené týmito technikami.

### 1.1 Optimalizačný problém

Optimalizačný problém (OP) všeobecne znamená nájsť riešenie úlohy

$$\min\{f(\mathbf{x}) : \mathbf{x} \in X\}, \text{ kde } f : X \rightarrow \mathbb{R} \text{ a } X \subset \mathbb{R}^n. \quad (1.1)$$

V texte budeme  $f$  nazývať *účelovou funkciou* a  $X$  *prípustnou množinou*. Ak je množina  $X$  prázdna, hovoríme, že optimalizačný problém je *neprípustný*. Ďalej, ak existuje postupnosť  $\mathbf{x}^k \in X$  taká, že  $f(\mathbf{x}^k) \rightarrow \infty$  pre  $k \rightarrow \infty$ , hovoríme o *neohraničenom probléme*. Pre úplnosť uvedieme nasledujúce definície:

**Definícia 1** Budeme hovoriť, že funkcia  $f : X \rightarrow \mathbb{R}$  má v bode  $\mathbf{x}$

i) **lokálne minimum** na  $X \subset \mathbb{R}^n$ , ak existuje  $\delta > 0$  také, že pre každé  $\mathbf{y} \in X$ ,  $\|\mathbf{y} - \mathbf{x}\| < \delta$  platí  $f(\mathbf{x}) \leq f(\mathbf{y})$ .

ii) **ostré lokálne minimum** na  $X \subset \mathbb{R}^n$ , ak existuje  $\delta > 0$  také, že pre každé  $\mathbf{y} \in X$ ,  $0 < \|\mathbf{y} - \mathbf{x}\| < \delta$  platí  $f(\mathbf{x}) < f(\mathbf{y})$ .

iii) **globálne minimum** na  $X \subset \mathbb{R}^n$ , ak pre každé  $\mathbf{y} \in X$  platí  $f(\mathbf{x}) \leq f(\mathbf{y})$ .

iv) **ostré globálne minimum** na  $X \subset \mathbb{R}^n$ , ak pre každé  $\mathbf{y} \in X$ ,  $\mathbf{x} \neq \mathbf{y}$  platí  $f(\mathbf{x}) < f(\mathbf{y})$ .

**Definícia 2** Budeme hovoriť, že funkcia  $f : X \rightarrow \mathbb{R}$  má v bode  $\mathbf{x}$

i) **lokálne maximum** na  $X \subset \mathbb{R}^n$ , ak existuje  $\delta > 0$  také, že pre každé  $\mathbf{y} \in X$ ,  $\|\mathbf{y} - \mathbf{x}\| < \delta$  platí  $f(\mathbf{x}) \geq f(\mathbf{y})$ .

ii) **ostré lokálne maximum** na  $X \subset \mathbb{R}^n$ , ak existuje  $\delta > 0$  také, že pre každé  $\mathbf{y} \in X$ ,  $0 < \|\mathbf{y} - \mathbf{x}\| < \delta$  platí  $f(\mathbf{x}) > f(\mathbf{y})$ .

iii) **globálne maximum** na  $X \subset \mathbb{R}^n$ , ak pre každé  $\mathbf{y} \in X$  platí  $f(\mathbf{x}) \geq f(\mathbf{y})$ .

iv) **ostré globálne maximum** na  $X \subset \mathbb{R}^n$ , ak pre každé  $\mathbf{y} \in X$ ,  $\mathbf{x} \neq \mathbf{y}$  platí  $f(\mathbf{x}) > f(\mathbf{y})$ .

Bežne pri matematickom modelovaní optimalizačného problému je množina  $X$  určená explicitne sadou reálnych rovníc a nerovnic

$$X := \{\mathbf{x} : g_i(\mathbf{x}) = 0, i \in I, h_j(\mathbf{x}) \geq 0, j \in J\},$$

kde  $I$  a  $J$  sú indexové množiny. Mnoho faktorov vplyva na to, ako sa dá efektívne vyriešiť daný problém. Počet rovníc a nerovnic ktoré definujú množinu  $X$ ,  $|I| + |J|$  je zrejme jedným z nich. Dôležitý vplyv na náročnosť výpočtu majú vlastnosti funkcií  $f$ ,  $g_i$  a  $h_i$ , hlavne spojitosť, (ne)linearita, diferencovateľnosť, konvexita  $f$ , konvexita množiny  $X$  a podobné vlastnosti. Z tohto dôvodu vzniklo (a naďalej vznikajú) na riešenie optimalizačných úloh viacerô druhov algoritmov, podľa vhodnosti na jednotlivé typy úloh.

V nasledujúcich podkapitolách stručne oboznámime čitateľa s bežnými typmi optimalizačných problémov. Nebudeme ich preberať od teoretických základov, keďže to nie je cieľom tejto diplomovej práce. Podrobnejšie, konkrétne optimalizačné metódy preberieme v neskorších kapitolách, alebo odkážeme na vhodnú literatúru. Nakoniec ich budeme numericky ilustrovať.

### 1.1.1 Lineárne programovanie (LP)

Lineárne programovanie je jeden z najrozšírenejších problémov matematického programovania. Minimalizujeme (maximalizujeme) tu lineárnu účelovú funkciu a množina  $X$  je explicitne daná množinou lineárnych rovností a nerovností:

$$\begin{aligned} \min_{\mathbf{x}} \mathbf{c}^\top \mathbf{x} \\ \mathbf{a}_i^\top \mathbf{x} &= b_i, & i \in I \\ \mathbf{a}_j^\top \mathbf{x} &\geq b_j, & j \in J \\ \mathbf{x} &\geq \mathbf{0}, \end{aligned} \tag{1.2}$$

kde  $\mathbf{a}_i, \mathbf{a}_j, \mathbf{c} \in \mathbb{R}^n$ ,  $b_i, b_j \in \mathbb{R}$ ,  $i \in I, j \in J$  su dané a  $\mathbf{x} \in \mathbb{R}^n$ . Všeobecne však možno túto úlohu previesť na ekvivalentnú úlohu s lineárnymi rovnosťami použitím dodatočných premenných:

$$\begin{aligned} \min_{\mathbf{x}} \mathbf{c}^\top \mathbf{x} \\ \mathbf{Ax} &= \mathbf{b} \\ \mathbf{x} &\geq \mathbf{0}, \end{aligned} \tag{1.3}$$

kde  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b} \in \mathbb{R}^m$  su dané, a  $\mathbf{x} \in \mathbb{R}^n$  je vektorová premenná. Tomuto tvaru zápisu hovoríme *standardný tvar LP*.

Doposiaľ najlepšie známe a úspešne používané algoritmy na riešenie úloh LP sú simplexová metóda (simplex method) a metóda vnútorného bodu (interior-point method).

Moderné algoritmy nám umožňujú riešiť úlohy lineárneho programovania s veľkým počtom premenných. Vďaka dobrej implementácii a dostupnosti vhodných algoritmov, niektoré matematické problémy formulované ako nelineárne, môžeme po dotatočnej úprave linearizovať a riešiť pomocou lineárneho programovania. V 4. kapitole pohovoríme o probléme výberu optimálneho portfólia vzhľadom k niektorým mieram rizika práve pomocou lineárnej formulácie úlohy.

### 1.1.2 Kvadratické programovanie (QP)

Jeden z ďalších často používaných optimalizačných problémov. Účelová funkcia je kvadratická, ale ohraničenia sú lineárne. Pre štandardnú formu úlohy kvadratického programovania máme zápis:

$$\begin{aligned} \min_{\mathbf{x}} \quad & \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ & \mathbf{A} \mathbf{x} = \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}, \end{aligned} \tag{1.4}$$

kde  $\mathbf{A} \in \mathbb{R}^{m \times n}$ ,  $\mathbf{b} \in \mathbb{R}^m$ ,  $\mathbf{c} \in \mathbb{R}^n$ ,  $\mathbf{Q} \in \mathbb{R}^{n \times n}$  su známe, a  $\mathbf{x} \in \mathbb{R}^n$  je vektorová premenná. Ohraničenia v tomto modeli nie sú obmedzené len na rovnosti. Zápis môže obsahovať aj lineárne nerovnosti, ktoré však použitím doplnkových premenných prevedieme na rovnosti a dostaneme tak tento štandardný tvar. Bez újmy na všeobecnosti môžeme predpokladať, že matica  $\mathbf{Q}$  je symetrická (pretože  $\mathbf{x}^T \mathbf{Q} \mathbf{x} = \frac{1}{2} \mathbf{x}^T (\mathbf{Q} + \mathbf{Q}^T) \mathbf{x}$ ). Ak je matica  $\mathbf{Q}$  pozitívne semidefinitná, tak účelová funkcia je konvexná. Úloha kvadratického programovania odpovedá úlohe nájsť optimálne portfólio pri požadovanom očakávanom výnose s minimálnou smerodajnou odchýlkou portfólia. Matica  $\mathbf{Q}$  tu práve prezentuje kovariančnú maticu aktív a tá je pozitívne semidefinitná. Vhodnou numerickou metódou na riešení tohto problému s pozitívne semidefinitnou maticou  $\mathbf{Q}$  je metóda vnútorného bodu pre kvadratické programovanie o ktorej budeme hovoriť neskôr. Práve pomocou tejto metódy prevedieme optimalizovanie Markowitzovho portfólia, čo je vlastne úloha kvadratického programovania.

## 1.2 Stochastické optimalizačné algoritmy

V poslednej dobe veľmi populárne optimalizačné algoritmy. Pojem „stochastické“ tu znamená, že v optimalizačnom procese hľadania optimálneho riešenia prípustnej množiny, je určitým spôsobom implementovaný náhodný prvok (náhodne čísla). Nejedná sa teda o deterministické metódy a riešenie získané takýmto prístupom je v určitom zmysle náhodné. V mnohých prípadoch nemožno hovoriť ani o konvergencii takejto metódy. Napriek tomu si získali veľkú popularitu v širokom spektre technických a prírodných vied. Aj keď pre takto získané riešenie je skoro nemožné vysloviť nejaké tvrdenie, prax a veľké množstvo testov týchto metód ukázali ich výhody hlavne v rýchlosti a schopnosti zvládnuť náročné výpočetné problémy s komplikovanými ohraničeniami a účelovou funkciou. Je totiž niekedy vhodné akceptovať dosiahnuté riešenie, ktoré je v určitom zmysle dostatočne optimálne (pre nás uspokojujúce), ako najšť skutočné optimum (ak je to vôbec možné) za cenu veľkých časových nárokov.

Jedná sa o metódy relatívne mladé, vyvinuté prevažne v druhej polovici 20. storočia, pretože ich použitie je podmienené dostatočnou výpočetnou kapacitou počítačov. Ich vývoj



sa prakticky nezastavil a stále sa objavujú ich nové modifikácie, zlepšenia a svoje uplatnenie si našli aj vo finančnom rozhodovaní. Medzi takéto algoritmy patria aj *evolučné*, alebo *genetické algoritmy*, ktoré našli inšpiráciu v evolučnej biológii a genetike. V samostatnej kapitole popíšeme princíp fungovania vybraných metód týchto typov. Venovať sa budeme Nelder-Meadovej metóde, simulovanému taveniu a metóde diferenciálnych evolúcií. Pomocou týchto algoritmov budeme riešiť úlohy s nekonvexnou účelovou funkciou a nelineárnymi ohraňovaniami.

## 1.3 Finančný management

### 1.3.1 Problém optimálneho portfólia

Voľba optimálneho portfólia ako finančnej investície nie je len o vhodnom výbere aktív, ale aj o ich diverzifikácii s cieľom minimalizovať celkové riziko investície. Pod rizikom rozumieme neistotu v budúcom vývoji hodnoty týchto aktív. Riešenie tohto problému si vyžaduje pozornosť pri vytváraní matematického modelu, aby reálne opisoval situáciu. Dôležitý je prístup k meraniu rizika ako neistote spojenej s investíciou. V práci sa budeme venovať tomuto problému pri implementácii rôznych mier rizika. Uvedme ešte definíciu takzvaných koherentných mier rizika.

**Definícia 3** *Reálnu funkciu  $\phi$ , na priestore reálnych náhodných veličín  $\mathcal{P}$  budeme nazývať koherentnou mierou rizika, ak pre  $X, Y \in \mathcal{P}$  a  $\alpha \in \mathbb{R}$  platí:*

- (A1)  $X \leq Y$  skoro iste  $\Rightarrow \phi(X) \leq \phi(Y)$  skoro iste (monotónnosť)
- (A2)  $\phi(X + Y) \leq \phi(X) + \phi(Y)$  (subaditivita)
- (A3)  $\phi(\alpha X) = \alpha \phi(X)$  (pozitívna homogenita)
- (A4)  $\phi(\alpha + X) = \alpha + \phi(X)$  (translačná ekvivariancia).

Individuálne aktívum má v čase zväčša iný vývoj (jeho hodnoty), ako vývoj hodnoty portfólia zloženého z viacerých inštrumentov. Naším cieľom preto bude zostrojiť toto portfólio tak, aby sme maximalizovali očakávaný výnos a minimalizovali riziko v poňatí celého portfólia. V definícii náhodné veličiny  $X$  a  $Y$  predstavujú straty dvoch aktív. Vhodnou mierou týchto strát (rizík) je intuitívne funkcia spĺňajúca axiómy definície. Axióma (A2) subaditivity, nám pri takto zvolenej miere rizika napovedá o diverzifikácii portfólia. Ako uvidíme neskôr, niektoré používané miery rizika nespĺňajú túto podmienku, avšak vďaka dobrej implementácii sú široko používané.

### 1.3.2 Ohraničenia v optimalizácii

Reálne obmedzenia účastníkov trhu, premietané do optimalizačných modelov sú významnou súčasťou optimalizačných metód. Do problému prinášajú zväčša ďalšie problémy. Ohraničenia v štandardizovaných modeloch matematického programovania zvyčajne nepostačujú k popisu reálnych situácií a musia byť dodatočne rozšírené, čo komplikuje tvar prípustnej množiny. Takéto úlohy je väčšinou nemožné alebo neefektívne symbolicky riešiť. V numerických algoritmoch možno problém transformovať na neohraňovaný použitím penalizačných funkcií, alebo iterované kroky viesť prípustnou množinou (metódy vnútorných bodov). Nech všeobecný optimalizačný problém je zapísaný v nasledujúcom tvare:

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) \\ \mathbf{x} \in A, \end{aligned} \tag{1.5}$$

kde  $A \subset X \subset \mathbb{R}^n$ . Tento problém môže byť reformulovaný na nasledujúci problém:

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) + p(d(\mathbf{x}, A)) \\ \mathbf{x} \in X \subset \mathbb{R}^n, \end{aligned} \tag{1.6}$$

kde  $d(\mathbf{x}, A)$  je funkcia, popisujúca vzdialenosť bodu  $\mathbf{x}$  od množiny  $A$  (alebo porušenie ohraničení) a  $p$  je monotónna, neklesajúca funkcia taká, že  $p(0) = 0$ . Funkcia  $p$  je obvykle nazývaná penalizačnou funkciou. V niektorých prípadoch môžu byť tieto úlohy ekvivalentné, alebo úlohou 1.6 možno získať horný odhad optimálneho riešenia. Podrobnejšie o použití penalizačných funkcií čitateľ nájde napríklad v [1].

V praxi najbežnejšie, napríklad transakčné náklady, alebo daňový systém môžu zásadne meniť rozhodovanie, alebo voľbu optimálnej stratégie investora. Niektoré finančné inštitúcie, v prípade umiestnenia finančných aktív sú priamo obmedzované zákonom. Jedná sa hlavne o penzijné fondy, poisťovne a banky, ktoré musia vykazovať požadovanú solventnosť. Obmedzenia sa hlavne týkajú umiestnenia v rizikových cenných papieroch. Požiadavky sú kladené napríklad na časť hodnoty majetku investovaného do určitej skupiny cenných papierov, alebo emitentov. Emitenti sa rozlišujú napríklad podľa geografického začlenenia a typu činností. Podľa typu sa tak najčastejšie rozlišujú štáty, municipality (verejná správa) a korporáty (firmy). V tomto smere je ešte dôležité hodnotenie emitenta ratingovými agentúrami. V rámci geografického rozlíšenia v našich pomeroch (Európska únia) je dôležité kritérium sídlo, respektíve či subjekt sídli v Európskom hospodárskom priestore. V prípade cenného papiera sa hodnotí jeho samotná rizikovosť, typ, poprípade druh jeho emitenta. Najčastejšie sa ohraničenia vzťahujú na dlhopisy, akcie, deriváty a na peňažnú menu v ktorej boli emitované. Dôležité sú takisto kritéria, ktoré sa stanovujú v rámci firmy a vstupujú do interných modelov.

# Kapitola 2

## Optimalizačné metódy všeobecne

### 2.1 Lineárne programovanie

Počiatky lineárneho programovania ako matematického modelu začali v priebehu druhej svetovej vojny so snahou optimalizovať vojenské procesy. Prvé publikácie na túto tému sa preto objavili až v prvých povojnových rokoch a spolu so vznikom prvých počítačov nastal ich širší vývoj. Za zakladateľa teórie je považovaný americký matematik George Bernard Dantzig, ktorý v roku 1947 formuloval úlohu lineárneho programovania. Na ďalšom vývoji mali výrazný podiel osobnosti ako Leonid Kantorovič, John von Neumann, Wassily Leontief a Tjalling Koopmans. Lineárne programovanie stálo všeobecne na začiatku matematického programovania a časom sa formulovali problémy kvadratického, stochastického a ďalšie typy matematického programovania. Viac zaujímavostí z vývoja tohto oboru sa čitateľ dozvie napríklad v [7].

#### 2.1.1 Simplexový algoritmus

Simplexový algoritmus, alebo simplexová metóda je prvý algoritmus, ktorý bol predstavený na riešenie úlohy lineárneho programovania. Vyvinul ju v roku 1947 George B. Dantzig. Táto metóda bola dokonca časopisom *The Journal Computing in Science and Engineering* zaradená v zozname najlepších 10 algoritmov 20. storočia. V texte nebudeme detailne popisovať simplexový algoritmus a k nemu potrebnú teóriu. Čitateľ nájde podrobný teoretický aparát napríklad v [4].

Stručne pripomeňme základný princíp simplexovej metódy. Uvažujme úlohu lineárneho programovania v štandardnom tvare (1.3). Predpokladajme, že maticu  $\mathbf{A} \in \mathbb{R}^{m \times n}$  môžeme zapísať v tvare  $\mathbf{A} = [\mathbf{B}, \mathbf{N}]$ , kde  $\mathbf{B} \in \mathbb{R}^{m \times m}$  je matica s hodnotou  $m$ . Podobne preznačme vektory  $\mathbf{x} = [\mathbf{x}_B, \mathbf{x}_N]$  a  $\mathbf{c} = [\mathbf{c}_B, \mathbf{c}_N]$ . Po krátkych úpravách, dostaneme ekvivalentný zápis pre ohraničenia:

$$\mathbf{x}_B + \mathbf{B}^{-1}\mathbf{N}\mathbf{x}_N = \mathbf{B}^{-1}\mathbf{b}.$$

Riešenie ktoré splňa  $\mathbf{x}_N = \mathbf{0}$  a  $\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}$  nazývame *bázické riešenie*. Ak ďalej  $\mathbf{x}_B \geq \mathbf{0}$  nazývame ho *bázickým prípustným riešením*. Označme  $\mathbf{Z} = \mathbf{c}^\top \mathbf{x}$  účelovú funkciu. Potom sú nasledujúce zápisy ekvivalentné:

$$\begin{aligned} \mathbf{Z} &= \mathbf{c}^\top \mathbf{x} \\ \mathbf{Z} - (\mathbf{c}_N - \mathbf{c}_B \mathbf{B}^{-1} \mathbf{N}) \mathbf{x}_N &= \mathbf{c}_B \mathbf{B}^{-1} \mathbf{b}. \end{aligned} \tag{2.1}$$

Dôležitý poznatok v teórii lineárneho programovania je, že ak úloha LP má optimálne riešenie, musí mať optimálne bázické prípustné riešenie. Takže pri hľadaní optimálneho riešenia, stačí skúmať tieto prípustné bázické riešenia. Takých riešení je konečne veľa, viď napríklad v [4]. Simplexový algoritmus nám dáva návod ako prípustné bázické riešenia prechádzať, pričom postupne znižovať hodnotu účelovej funkcie. Význam poslednej rovnosti v (2.1) spočíva v tom, že ak  $\mathbf{c}_N - \mathbf{c}_B \mathbf{B}^{-1} \mathbf{N} \leq \mathbf{0}$ , tak bázické riešenie  $\mathbf{x}_B = \mathbf{B}^{-1} \mathbf{b}$ ,  $\mathbf{x}_N = \mathbf{0}$  je optimálne riešenie.

Simplexový algoritmus vždy nájde optimálne riešenie (ak existuje), ale dôležitou otázkou je, aké výpočetné nároky tento algoritmus potrebuje. Dlhú dobu nebolo známe, či simplexový algoritmus patrí k nepolynomiálnej alebo polynomiálnej triede algoritmov. V roku 1970, Victor Klee a George Minty vytvorili príklad úlohy LP, na ktorej vyriešenie potrebuje simplexový algoritmus exponenciálny počet krokov v najhoršom prípade. V roku 1978 ruský matematik Leonid Khachiyan dokázal, že *elipsoidný algoritmus* (typ subgradientnej metódy), rieši LP v menej ako polynomiálnom čase v závislosti na rozmeroch matice  $\mathbf{A}$  a počtom číslíc na vstupe. Napriek tomu sa tento algoritmus ukázal ako neefektívny pre praktické problémy.

V roku 1984 indický matematik Narendra Karmarkar predstavil algoritmus *vnútorného bodu* a dokázal, že úlohu LP rieši v polynomiálnom čase. Tento algoritmus na rozdiel od elipsoidného bol efektívny pre praktické úlohy s vysokým počtom premenných. Tomuto typu algoritmu sa budeme venovať aj v súvislosti s kvadratickým programovaním a tiež uvedieme aj formu tohto algoritmu pre lineárne programovanie. Viac z histórie lineárneho programovania sa čitateľ dozvie v [7].

## 2.2 Nelineárne programovanie

Nelineárne programovanie sa začalo rozvíjať okolo roku 1951 so známymi Karush-Kuhn-Tuckerovými podmienkami optimality (KKT). Pod nelineárnym optimalizačným problémom rozumieme nasledujúcu úlohu:

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) \\ g_i(\mathbf{x}) = 0, \quad i \in \mathcal{I} \\ h_i(\mathbf{x}) \geq 0, \quad i \in \mathcal{J}, \end{aligned}$$

kde  $f$  a  $g$  sú reálne funkcie  $\mathbb{R}^n \rightarrow \mathbb{R}$  a  $\mathcal{I}$ ,  $\mathcal{J}$  sú indexové množiny.

Na riešenie nelineárnych úloh sa najčastejšie používajú numerické algoritmy, ktoré môžeme zo širšieho hľadiska kategorizovať do dvoch základných skupín, a to *gradientné metódy* a *priame metódy*. Gradientné metódy využívajú informáciu prvej a druhej derivácie, jedná sa napríklad o metódy *vnútorného bodu*. Priame metódy primárne nevyužívajú informáciu derivácie a patria tu napríklad *genetické* alebo *stochastické* algoritmy. Týmto metódam sa budeme venovať v 3. kapitole.

### 2.2.1 Gradientné metódy

V tejto časti predstavíme niektoré základné numerické techniky riešenia rovníc a metód optimalizácie. Sú jednoduché, avšak dôležité a často využívané v zložitejších optimalizačných algoritmoch.

## Binárne hľadanie (Binary search)

Binárne hľadanie je založené na veľmi jednoduchej myšlienke na riešenie rovnice jednej premennej  $f(x) = 0$ , kde  $f$  je spojitá funkcia. Často  $x$  nemožno explicitne vyjadriť, preto je potrebné riešiť rovnicu numericky. Najprv nájdeme dva body  $a, b$  také, že  $f(a), f(b)$  majú opačné znamienka. Keďže  $f$  je spojitá, tak vieme že v intervale  $(a, b)$  je riešenie. Môžeme teraz povedať, že máme interval spoľahlivosti  $(a, b)$ . V ďalšom kroku zrátame hodnotu  $f(\frac{a+b}{2})$ , a podľa znamienka tejto hodnoty rozhodneme o ďalšom intervale. Buď  $(a, \frac{a+b}{2})$ , alebo  $(\frac{a+b}{2}, b)$ , tak aby funkčné hodnoty v krajných bodoch mali opačné znamienka. Tento postup na novom intervale opakujeme.

Binárne hľadanie je veľmi rýchle. Po  $k$  krokoch máme interval spoľahlivosti o dĺžke  $(b - a)2^{-k}$ . Nedostatkom tohto prístupu je, že algoritmus nájde len jedno riešenie. Binárne hľadanie môžeme vo financiách využiť napríklad pri hľadaní *vnútornej miery výnosnosti (internal rate of return)* investície.

## Metóda najväčšieho poklesu (Steepest Descent)

Najjednoduchšia numerická metóda ako nájsť minimum funkcie bez ohraničení je založená na myšlienke hľadania optima v smere klesania funkcie. Ak je funkcia  $f$  diferencovateľná, tak jej záporný gradient  $-\nabla f(\mathbf{x})$  udáva smer najväčšieho klesania funkcie  $f$  v bode  $\mathbf{x}$ . Smer klesania máme, takže zostáva určiť veľkosť kroku v tomto smere. Predpokladajme, že počiatočný odhad riešenia je  $\mathbf{x}^0$  a označme  $\mathbf{d}^0 = -\nabla f(\mathbf{x}^0)$ . Nový bod v smere  $\mathbf{d}^0$  o veľkosti  $\alpha$  je  $\mathbf{x}^0 + \alpha\mathbf{d}^0$ . Označme

$$\phi(\alpha) := f(\mathbf{x}^0 + \alpha\mathbf{d}^0).$$

Optimálna veľkosť kroku  $\alpha$  je pri minimálnej hodnote  $\phi(\alpha)$ . Namiesto zráťania presnej hodnoty  $\alpha$ , zrátame len aproximáciu. Chceme také  $\alpha$ , ktoré splňuje podmienku

$$\phi(\alpha) \leq \phi(0) + \mu\phi'(0),$$

kde  $\mu \in (0, 1)$  je zvolená konštanta. Túto podmienku nazývame *Armijo-Goldsteinova podmienka*. Ukázalo sa, že odhad splňajúci túto podmienku je pre iteráciu v algoritme postačujúci, oproti presnému riešeniu, ktoré je v mnohých prípadoch náročnejšie zrátať. Kritérium na ukončenie iterácie môžeme zvoliť viacerými spôsobmi. Keďže gradient funkcie  $f$  musí byť v optime 0, tak vhodným kritériom je  $\|\nabla f(\mathbf{x})\| \leq \varepsilon$  pre vhodne zvolené  $\varepsilon > 0$ . Podobne, vhodným kritériom je, ak iterované riešenia začnú byť dostatočne blízko:  $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| \leq \varepsilon$ , pre nejaké  $\varepsilon > 0$ .

Tento algoritmus je však dosť pomalý. Smery najväčšieho poklesu v iteráciách menia znamienko, smer najväčšieho klesania je totiž kolmý na vrstevnice funkcie. Iterované riešenia alebo smery iterácii sú preto dosť kľukaté (zig-zaging), čo vysvetľuje pomalosť tejto metódy.

## Newtonova metóda

Jedná sa o veľmi starú a často používanú numerickú metódu na riešenie rovníc vyvinutú Isaacom Newtonom. Predpokladajme, že hľadáme riešenie systému spojito diferencovateľ-

ných rovníc:

$$\begin{aligned}f_1(x_1, x_2, \dots, x_n) &= 0 \\f_2(x_1, x_2, \dots, x_n) &= 0 \\&\vdots \\f_n(x_1, x_2, \dots, x_n) &= 0.\end{aligned}$$

Označme tento systém zápisom

$$F(\mathbf{x}) = \mathbf{0},$$

kde  $\mathbf{x} \in \mathbb{R}^n$ . Nech  $\mathbf{x}^k$  je súčasný odhad riešenia sústavy. Použitím Taylorovho rozvoja funkcie  $F$  okolo  $\mathbf{x}^k$  dostaneme

$$F(\mathbf{x}^k + \delta) \approx \hat{F}(\delta) := F(\mathbf{x}^k) + \nabla F(\mathbf{x}^k)\delta, \quad (2.2)$$

kde  $\nabla F(x)$  je Jakobián funkcie  $F$  a  $\hat{F}(\delta)$  je lineárna aproximácia funkcie  $F$  okolo bodu  $\mathbf{x}^k$ . V nasledujúcom kroku hľadáme hodnotu  $\delta$  takú, aby

$$F(\mathbf{x}^k) + \nabla F(\mathbf{x}^k)\delta = \mathbf{0}.$$

Ak je matica  $\nabla F(\mathbf{x}^k)$  regulárna, tak táto rovnica má jednoznačné riešenie.

$$\delta = -\nabla F(\mathbf{x}^k)^{-1}F(\mathbf{x}^k)$$

Takže v tomto prípade pre nové riešenie  $\mathbf{x}^k$  máme:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \delta = \mathbf{x}^k - \nabla F(\mathbf{x}^k)^{-1}F(\mathbf{x}^k).$$

Hľadáme teraz minimum reálnej, dvakrát spojitou diferencovateľnej funkcie  $f(\mathbf{x})$ ,  $\mathbf{x} \in \mathbb{R}^n$ . Nutná podmienka pre optimum je  $\nabla f(\mathbf{x}) = \mathbf{0}$ . Označme parciálne derivácie funkcie  $f(\mathbf{x})$

$$f_i(\mathbf{x}) = \frac{\partial f}{\partial x_i}(x_1, \dots, x_n).$$

Takže potrebujeme, aby

$$\nabla F(\mathbf{x}) = \nabla^2 f(\mathbf{x}).$$

Ak súčasný odhad je  $\mathbf{x}^k$ , nasledujúci krok v iterácii bude  $\delta = -\nabla^2 f(\mathbf{x}^k)^{-1}\nabla f(\mathbf{x}^k)$ . Pre nový odhad optima máme

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \delta = \mathbf{x}^k - \nabla^2 f(\mathbf{x}^k)^{-1}\nabla f(\mathbf{x}^k).$$

Napriek tomu, že Newtonova metóda má výborné konvergenčné vlastnosti blízko k riešeniu, niekedy ju nie je vhodné použiť. Hlavne pri úlohách o veľkom počte neznámych je veľmi náročné spočítať Hesián v každej iterácii. V takýchto prípadoch namiesto presného Hesiána zrátame jeho aproximáciu tak, že to je výpočtovo menej náročné. Metódy s týmto prístupom sú známe ako *kvázi-newtonove metódy*.

## 2.3 Metódy vnútorného bodu (Interior-Point Methods)

Metódy vnútorného bodu (IPM) sa začali objavovať v 60. rokoch minulého storočia. Pôvodne boli navrhnuté k riešeniu LP úloh, ale neskôr sa ukázali vhodné aj k riešeniu úloh iných typov. Na rozdiel od simplexovej metódy, iterácie prebiehajú vnútri prípustnej množiny, a nie po krajných bodoch. IPM je veľmi dobrý nástroj pre úlohy lineárneho programovania, ale tiež sa ukázalo, že dokáže riešiť aj úlohy iných typov, napríklad kvadratického programovania. Úlohu konvexného kvadratického programovania rieši dokonca v polynomiálnom čase. Dá sa povedať, že IPM je modifikáciou Newtonovej metódy s tým, že dokáže pracovať s ohraničeniami.

### 2.3.1 Metóda vnútorného bodu pre konvexné QP

V tejto časti predstavíme metódu vnútorného bodu pre konvexné kvadratické programovanie a stručný popis pre lineárne programovanie. Je založená na podmienkach optimality pre QP a niekedy je nazývaná aj ako primárne-duálna metóda vnútorného bodu (primal-dual interior point method). Pripomeňme Karush-Kuhn-Tuckerove podmienky optimality, špeciálne pre úlohu 1.4:

**Tvrdenie 1** *Uvažujme úlohu kvadratického programovania (1.4), kde  $\mathbf{Q}$  je pozitívne semidefinítaná matica. Potom  $\mathbf{x} \geq \mathbf{0}$  je globálne optimum úlohy QP práve vtedy, ak existujú vektory  $\mathbf{s} \in \mathbb{R}^n$  a  $\mathbf{y} \in \mathbb{R}^m$  také, že platí*

$$\begin{aligned} \mathbf{A}^\top \mathbf{y} - \mathbf{Q}\mathbf{x} + \mathbf{s} &= \mathbf{c} \\ \mathbf{x} &\geq \mathbf{0}, \mathbf{s} \geq \mathbf{0} \\ x_i s_i &= 0, \quad i = 1, \dots, n. \end{aligned} \tag{2.3}$$

Dôkaz, všeobecne pre úlohu nelineárneho programovania nájde čitateľ v [4]. Zapišme tieto podmienky pomocou funkcie  $F$ :

$$F(\mathbf{x}, \mathbf{y}, \mathbf{s}) = \begin{bmatrix} \mathbf{A}^\top \mathbf{y} - \mathbf{Q}\mathbf{x} + \mathbf{s} - \mathbf{c} \\ \mathbf{A}\mathbf{x} - \mathbf{b} \\ \mathbf{X}\mathbf{S}\mathbf{e} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad (\mathbf{x}, \mathbf{s}) \geq \mathbf{0}, \tag{2.4}$$

kde  $\mathbf{X}$  a  $\mathbf{S}$  sú diagonálne matice s prvkami vektorov  $\mathbf{x}$  a  $\mathbf{s}$  na diagonále a  $\mathbf{e}$  je vektor jednotiek príslušnej veľkosti.

Systém rovníc  $F(\mathbf{x}, \mathbf{y}, \mathbf{s}) = \mathbf{0}$  má  $n+m+n$  neznámych. Pretože je nelineárny, nemôžeme ho riešiť použitím Gaussovej eliminačnej metódy. Systém je však kvadratický, takže vhodné by bolo použiť Newtonovu metódu, problém však robia ohraničenia nezápornosti.

Princíp metódy vnútorného bodu spočíva v tom, že použijeme modifikovanú Newtonovu metódu. Najprv zostrojíme počiatočné riešenie  $(\mathbf{x}^0, \mathbf{y}^0, \mathbf{s}^0)$ , ktoré vyhovuje prvým dvom podmienkam v rovnosti  $F(\mathbf{x}, \mathbf{y}, \mathbf{s}) = \mathbf{0}$  a podmienkam nezápornosti  $\mathbf{x}^0 > \mathbf{0}$ ,  $\mathbf{s}^0 > \mathbf{0}$ . Zrejme  $(\mathbf{x}^0, \mathbf{y}^0, \mathbf{s}^0) \in \mathcal{F}^\circ$ , kde

$$\mathcal{F}^\circ := \{(\mathbf{x}, \mathbf{y}, \mathbf{s}) : \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{A}^\top \mathbf{y} - \mathbf{Q}\mathbf{x} + \mathbf{s} = \mathbf{c}, \mathbf{x} > \mathbf{0}, \mathbf{s} > \mathbf{0}\}. \tag{2.5}$$

Definujme ešte množinu  $\mathcal{F}$ :

$$\mathcal{F} := \{(\mathbf{x}, \mathbf{y}, \mathbf{s}) : \mathbf{A}\mathbf{x} = \mathbf{b}, \mathbf{A}^\top \mathbf{y} - \mathbf{Q}\mathbf{x} + \mathbf{s} = \mathbf{c}, \mathbf{x} \geq \mathbf{0}, \mathbf{s} \geq \mathbf{0}\}. \tag{2.6}$$

Následne sa pokúsime generovať nové riešenia  $(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) \in \mathcal{F}^o$ , postupne tak, aby sa blížili splneniu tretej podmienky v rovnosti  $F(\mathbf{x}, \mathbf{y}, \mathbf{s}) = \mathbf{0}$ . Predpokladajme teraz, že máme súčasný odhad riešenia  $(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k)$ . Zostrojíme nové riešenie užitím Newtonovej metódy:

$$J_F(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) \begin{bmatrix} \Delta \mathbf{x}^k \\ \Delta \mathbf{y}^k \\ \Delta \mathbf{s}^k \end{bmatrix} = -F(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k), \quad (2.7)$$

kde  $J_F(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k)$  je Jakobián funkcie  $F$  a  $[\Delta \mathbf{x}^k, \Delta \mathbf{y}^k, \Delta \mathbf{s}^k]$  je hľadaný smer nového riešenia. Jakobián je:

$$J_F(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) = \begin{bmatrix} -\mathbf{Q} & \mathbf{A}^\top & \mathbf{I} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}^k & \mathbf{0} & \mathbf{X}^k \end{bmatrix}. \quad (2.8)$$

Navyše, ak  $(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) \in \mathcal{F}^o$ , tak

$$F(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{X}^k \mathbf{S}^k \mathbf{e} \end{bmatrix} \quad (2.9)$$

a rovnica v Newtonovom algoritme pre nové riešenie je

$$\begin{bmatrix} -\mathbf{Q} & \mathbf{A}^\top & \mathbf{I} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}^k & \mathbf{0} & \mathbf{X}^k \end{bmatrix} \begin{bmatrix} \Delta \mathbf{x}^k \\ \Delta \mathbf{y}^k \\ \Delta \mathbf{s}^k \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ -\mathbf{X}^k \mathbf{S}^k \mathbf{e} \end{bmatrix}. \quad (2.10)$$

V našom prípade ale takto zrátané nové riešenie nemusí spĺňať podmienky  $\mathbf{s} \geq \mathbf{0}$  a  $\mathbf{x} \geq \mathbf{0}$ . Jednou z možností je zvoliť parameter  $\alpha_k \in (0, 1]$  taký, aby platilo  $\mathbf{x}^k + \alpha_k \Delta \mathbf{x}^k > \mathbf{0}$  a  $\mathbf{s}^k + \alpha_k \Delta \mathbf{s}^k > \mathbf{0}$ . Toto sa však ukázalo dosť neefektívne. Prípustné hodnoty pre parameter  $\alpha^k$  sú často malé a znamenajú malý posun k optimálnemu riešeniu, čo je dosť neefektívne. Ak nové riešenie iterujeme týmto spôsobom, hovoríme o *obyčajnom Newtonovom kroku*. Nové riešenie preto konštruujeme v okolí takzvaného *centrálneho ťahu*. Centrálnym ťahom označujeme trajektóriu (množinu)  $\mathcal{C}$  vnútri množiny  $\mathcal{F}^o$ . Definujeme ju pomocou parametra  $\tau > 0$  nasledujúcim spôsobom: bod  $(\mathbf{x}_\tau, \mathbf{y}_\tau, \mathbf{s}_\tau) \in \mathcal{C}$  ak vyhovuje rovnostiam

$$F(\mathbf{x}_\tau, \mathbf{y}_\tau, \mathbf{s}_\tau) = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \tau \mathbf{e} \end{bmatrix}, \quad (\mathbf{x}_\tau, \mathbf{s}_\tau) > \mathbf{0}. \quad (2.11)$$

Centrálny ťah  $\mathcal{C}$  potom definujeme:

$$\mathcal{C} := \{(\mathbf{x}_\tau, \mathbf{y}_\tau, \mathbf{s}_\tau) : \tau > 0\}. \quad (2.12)$$

Inými slovami, od  $\mathbf{x}$  a  $\mathbf{s}$  už nevyžadujeme komplementaritu (súčin  $\mathbf{0}$ ), ale to, že ich súčin v zložkách je  $\tau$ . Ak je množina  $\mathcal{F}^o$  neprázdna, tak systém (2.11) má pre  $\tau > 0$  jednoznačné riešenie  $(\mathbf{x}_\tau, \mathbf{y}_\tau, \mathbf{s}_\tau)$ . To znamená, že pre  $\tau \rightarrow \infty$ , riešenie  $(\mathbf{x}_\tau, \mathbf{y}_\tau, \mathbf{s}_\tau)$  konverguje k optimálnemu riešeniu. Pokúsme sa teraz Newtonovým ťahom iterovať nové riešenie v smere



centrálneho ľahu. Takto zostrojené nové riešenie „smeruje“ do množiny  $\mathcal{F}^o$  popri centrálnom ľahu, a môžeme očakávať, že pri znižovaní  $\tau$  sa bude blížiť k optimálnemu riešeniu. Iteráciu v smere centrálného ľahu nazývame *centrovaný Newtonov krok*. Takže opäť, nech  $(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k)$  je súčasný odhad riešenia. Body na centrálnom ľahu vyhovujú rovnosti (2.11). Označením pomocou funkcie  $\hat{F}$  je to

$$\hat{F}(\mathbf{x}, \mathbf{y}, \mathbf{s}) := \begin{bmatrix} \mathbf{A}^\top \mathbf{y} - \mathbf{Q}\mathbf{x} + \mathbf{s} - \mathbf{c} \\ \mathbf{A}\mathbf{x} - \mathbf{b} \\ \mathbf{X}\mathbf{S}\mathbf{e} - \tau\mathbf{e} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}. \quad (2.13)$$

Keďže platí  $J_F(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) = J_{\hat{F}}(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k)$ , smer centrovaného Newtonovho kroku  $(\Delta\mathbf{x}_c^k, \Delta\mathbf{y}_c^k, \Delta\mathbf{s}_c^k)$  dostaneme riešením systému:

$$\begin{bmatrix} -\mathbf{Q} & \mathbf{A}^\top & \mathbf{I} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}^k & \mathbf{0} & \mathbf{X}^k \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}_c^k \\ \Delta\mathbf{y}_c^k \\ \Delta\mathbf{s}_c^k \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \tau\mathbf{e} - \mathbf{X}^k\mathbf{S}^k\mathbf{e} \end{bmatrix}. \quad (2.14)$$

Upresníme ešte, ako volíme v priebehu algoritmu parameter  $\tau$ . Z tohto dôvodu definujeme mieru *diferencie duality* (*duality gap*)  $\mu$ :

$$\mu = \mu(\mathbf{x}, \mathbf{s}) := \frac{\sum_{i=1}^n x_i s_i}{n} = \frac{\mathbf{x}^\top \mathbf{s}}{n}. \quad (2.15)$$

Ak je bod  $(\mathbf{x}, \mathbf{y}, \mathbf{s})$  prípustný, teda  $(\mathbf{x}, \mathbf{y}, \mathbf{s}) \in \mathcal{F}$ , tak  $(\mathbf{x}, \mathbf{y}, \mathbf{s})$  je optimálne riešenie práve vtedy, ak  $\mu(\mathbf{x}, \mathbf{s}) = 0$ . Diferencia duality  $\mu$  je dobrá miera optimality, čím menšie  $\mu$ , tým sme bližšie optimu. Predpokladajme, že súčasnému riešeniu  $(\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k)$  prislúcha hodnota  $\mu^k := \mu(\mathbf{x}^k, \mathbf{s}^k)$ . Ak teraz iterujeme centrovaný Newtonov krok, tak musíme zvoliť cieľené  $\tau$  pre nové riešenie. Všeobecný postup je zvolenie  $\tau^k := \mu^k \sigma^k$ , kde  $\sigma^k \in [0, 1]$ . Voľba parametra  $\sigma^k$  nám dáva ďalšie možnosti v ovplyvnení priebehu algoritmu.

Nasledujúca schéma popisuje základný algoritmus IPM:

### Základný algoritmus metódy vnútorného bodu pre konvexné QP

1. Zvolíme  $(\mathbf{x}^0, \mathbf{y}^0, \mathbf{s}^0) \in \mathcal{F}^o$ .
2. Zvolíme  $\sigma \in [0, 1]$  a nech  $\mu^k = \frac{(\mathbf{x}^k)^\top \mathbf{s}^k}{n}$ . Riešime

$$\begin{bmatrix} -\mathbf{Q} & \mathbf{A}^\top & \mathbf{I} \\ \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{S}^k & \mathbf{0} & \mathbf{X}^k \end{bmatrix} \begin{bmatrix} \Delta\mathbf{x}^k \\ \Delta\mathbf{y}^k \\ \Delta\mathbf{s}^k \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \sigma^k \mu^k \mathbf{e} - \mathbf{X}^k \mathbf{S}^k \mathbf{e} \end{bmatrix}.$$

3. Zvolíme  $\alpha^k$  také, že

$$\mathbf{x}^k + \alpha^k \Delta\mathbf{x}^k > \mathbf{0}, \text{ a } \mathbf{s}^k + \alpha^k \Delta\mathbf{s}^k > \mathbf{0}.$$

Nastav

$$(\mathbf{x}^{k+1}, \mathbf{y}^{k+1}, \mathbf{s}^{k+1}) = (\mathbf{x}^k, \mathbf{y}^k, \mathbf{s}^k) + \alpha^k (\Delta\mathbf{x}^k, \Delta\mathbf{y}^k, \Delta\mathbf{s}^k), \text{ a}$$

$k = k + 1$ .

Vo voľbe použitia parametrov  $\sigma^k$  a  $\tau^k$  vzniklo viacero variánt metódy vnútorného bodu, u ktorých táto voľba má vplyv na konvergenčné vlastnosti algoritmu. Niekedy sa zdá, že uvedený postup aproximovania bodov na centrálnom ťahu je zbytočne zložitý. Avšak tento prístup nám dáva možnosť v reálnom čase nájsť vhodné priblíženie optimálnemu riešeniu. Zrátanie presného riešenia systému podmienok optimality, alebo bodov na centrálnom ťahu je ďaleko obtiažnejšie. V algoritme ešte treba upresniť, kedy je dosiahnuté riešenie považované za dostatočne blízke optimálnemu riešeniu a teda iterácie ukončiť. Vhodné je k tomu použiť diferenciu duality  $\mu$ . Dosiahnuté riešenie v  $k$ -tej iterácii považujeme za dostatočne blízke optimálnemu, ak pre dosiahnuté riešenie platí:

$$\mu(\mathbf{x}^k, \mathbf{s}^k) \leq \varepsilon, \quad \varepsilon > 0,$$

kde  $\varepsilon$  je dostatočne malé zvolené číslo.

### 2.3.2 Metóda vnútorného bodu pre LP

Vráťme sa ešte na chvíľu k lineárnemu programovaniu. Už sme spomenuli, že metóda vnútorného bodu bola pôvodne vyvinutá pre úlohy lineárneho programovania. Aj keď je simplexová metóda výborný algoritmus, ktorý dosiahne presné optimum, pri úlohách väčšieho rozsahu je tento algoritmus neefektívny. Pri vysokom počte premenných je potrebné k zrátaniu výsledku veľký počet krokov, čo je časovo náročné. Metóda vnútorného bodu je hlavne pri úlohách väčších rozmerov podstatne rýchlejšia. Navyše takto zrátané riešenie môžeme dosiahnuť s dostatočne vysokou presnosťou. Úlohy takýchto, dokonca väčších rozmerov nie sú v praxi výnimkou. Aj v tomto prípade vzniklo pre lineárne úlohy viacej typov metód vnútorného bodu. Uvedieme tu základný popis, detaily nájde čitateľ napríklad v [14].

Uvažujme teda úlohu LP v štandardnom tvare (1.3) a jej optimálne riešenie  $\mathbf{x}^*$ . Tento problém môžeme riešiť užitím takzvanej logaritmickkej bariérovej (penalizačnej) funkcie. Uvažujme optimalizačnú úlohu:

$$\begin{aligned} \min \mathbf{c}^\top \mathbf{x} - t \sum_{i=1}^n \ln(x_i) \\ \mathbf{Ax} = \mathbf{b}, \quad t > 0, \end{aligned}$$

kde  $\mathbf{x} = (x_1, \dots, x_n)^\top$ . Označme riešenie týchto úloh pre  $t \geq 0$  symbolom  $\mathbf{x}(t)$ . Za vhodných predpokladov možno ukázať, že riešenia  $\mathbf{x}(t)$  existujú a pre  $t \rightarrow 0$  je  $\mathbf{x}(t) \rightarrow \mathbf{x}^*$ . Podobne, body  $\mathbf{x}(t)$ ,  $t \geq 0$  nám definujú množinu-centrálny ťah. Podmienka optimality KKT prvého rádu pre tento systém je:

$$\mathbf{c} - t\mathbf{X}^{-1}\mathbf{e} = \mathbf{A}^\top \mathbf{y}, \quad \mathbf{Ax} = \mathbf{b},$$

kde  $\mathbf{X}$  je diagonálna matica s prvkami vektora  $\mathbf{x}$  na diagonále. Položme  $\mathbf{z} = t\mathbf{X}^{-1}\mathbf{e}$ , jedná sa teda o systém  $2m + n$  lineárnych rovníc:

$$\begin{aligned} \mathbf{xz} &= t\mathbf{e} \\ \mathbf{A}^\top \mathbf{y} + \mathbf{z} &= \mathbf{c} \\ \mathbf{Ax} &= \mathbf{b}. \end{aligned}$$

K riešeniu tohto systému môžeme použiť Newtonovu metódu. Pre Newtonov krok  $(\Delta\mathbf{x}, \Delta\mathbf{y}, \Delta\mathbf{z})$  riešime systém

$$\begin{bmatrix} \mathbf{X} & \mathbf{Z} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{A}^\top \\ \mathbf{0} & \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \Delta\mathbf{z} \\ \Delta\mathbf{x} \\ \Delta\mathbf{y} \end{bmatrix} = \begin{bmatrix} t\mathbf{e} - \mathbf{xz} \\ \mathbf{c} - \mathbf{A}^\top \mathbf{y} - \mathbf{z} \\ \mathbf{b} - \mathbf{Ax} \end{bmatrix}.$$

System možno ešte ďalej upraviť použitím Gaussovej eliminačnej metódy na jednoduchší tvar. Podobne ako v modifikácii algoritmu pre kvadratické programovanie, parameter  $t$  postupne pri iteráciách znižujeme, až kým nedosiahneme vhodne zvolené zastavovacie kritérium. K voľbe zastavovacieho kritéria možno použiť informáciu z takzvanej *duálnej úlohy* k úlohe 1.3:

$$\begin{aligned} \max \mathbf{b}^\top \mathbf{y} \\ \mathbf{A}^\top \mathbf{y} + \mathbf{z} = \mathbf{c} \\ \mathbf{z} \geq \mathbf{0}, \mathbf{y} \in \mathbb{R}^n, \end{aligned}$$

ktorá nam dáva dolný odhad funkčnej hodnoty v optimálnom bode (detaily napríklad v [4]). Jedným z možných kritérií je, ak platí:

$$\frac{\|\mathbf{b} - \mathbf{A}\mathbf{x}\|}{\max(1, \|\mathbf{b}\|)} + \frac{\|\mathbf{c} - \mathbf{A}^\top \mathbf{y} - \mathbf{z}\|}{\max(1, \|\mathbf{c}\|)} + \frac{\|\mathbf{c}^\top \mathbf{x} - \mathbf{b}^\top \mathbf{y}\|}{\max(1, \|\mathbf{c}^\top \mathbf{x}\|, \|\mathbf{b}^\top \mathbf{y}\|)} \leq \varepsilon,$$

kde  $\varepsilon > 0$  je dostatočne malé zvolené číslo. S týmto kritériom pracuje napríklad systém Mathematica.

# Kapitola 3

## Vybrané optimalizačné metódy

V posledných desaťročiach nastal prudký rozvoj výpočetnej techniky a s tým spojený rozvoj výpočetných algoritmov schopných spracovávať veľké množstvo dát. V minulosti vzniklo mnoho metód schopných elegantne riešiť optimalizačné problémy a nájsť optimálne riešenie. Lenže tie sú často obmedzené na striktné predpoklady účelovej funkcie a ohraničení. Čo navyše, súčasné optimalizačné problémy vo financiách narážajú na problém veľkého množstva dát. Problémy o tisícoch premenných je časovo riešiť veľmi náročné. V praxi je niekedy potrebné získať riešenie v relatívne krátkom čase, čo je na spomínaný počet premenných neľahká úloha. Preto je často kladený dôraz na rýchlosť výpočtu, na úkor presnosti výsledku, ktorý ale v mnohých prípadoch je postačujúca informácia.

V nasledujúcej časti predstavíme niektoré optimalizačné metódy-výpočetné algoritmy, ktoré sú vďaka trendu výpočetných možností ambiciózne k riešeniu finančných optimalizačných úloh. Tieto metódy patria globálne do skupiny takzvaných heuristických (objaviteľských) algoritmov. To znamená, že algoritmus postupne hľadá (objavuje) lepšie riešenia. Výhodou oproti explicitným metódam je nenáročnosť na vstupné predpoklady a schopnosť poradiť si s lokálnymi extrémami. Nevýhodou je, že všeobecne neexistuje dôkaz ich konvergencie ku globálnemu optimu. Jednotlivé metódy a ich efektivita sa líšia pri riešení úloh rôznych typov. Budeme sa venovať trom typom týchto algoritmov: Nelder-Meadovej metóde, metóde diferenciálnych evolúcií a simulovanému taveniu.

### 3.1 Nelder-Meadova metóda (NM)

Nelder-Meadova metóda nesie vo svojom názve mená jej autoroch, Johna Neldera a R. Meada. Bola publikovaná v roku 1965. Niekedy je nazývaná aj ako *downhill simplex method*, alebo tiež *flexibilná metóda polyhedrónov*. Algoritmus pracuje na princípe ohodnotenia účelovej funkcie na vrcholoch simplexu a jeho následnou transformáciou na nový simplex, s menšími hodnotami účelovej funkcie v jeho vrcholoch.

**Definícia 4** Množinu  $A \subset \mathbb{R}^n$  nazývame konvexný polyéder, ak existuje konečná množina  $S \subset \mathbb{R}^n$  taká, že  $A = \text{conv}(S)$ .  $\text{conv}(S)$  označuje najmenšiu konvexnú množinu obsahujúcu množinu  $S$ .

**Definícia 5** Množinu  $S \subset \mathbb{R}^n$  nazývame simplex, ak je konvexný polyéder a každý bod  $S$  možno vyjadriť ako jednoznačne určenú lineárnu konvexnú kombináciu krajných bodov množiny  $S$ . Simplex  $S$  nazývame nedegenerovaný, ak množina jeho vnútorných bodov je neprázdna.

Z vlastností simplexu vyplýva, že simplex v  $\mathbb{R}^n$  má práve  $n + 1$  vrcholov. Viac o konvexnej analýze čitateľ nájde v [4].

V nasledujúcej podkapitole popíšeme NM algoritmus. Hľadáme globálne minimum funkcie  $f : \mathbb{R}^n \rightarrow \mathbb{R}$ .

### 3.1.1 Nelder-Meadov algoritmus

Tento algoritmus vytvára postupnosť transformovaných simplexov, s cieľom minimalizovať určitým spôsobom funkčné hodnoty v krajných bodoch simplexov. V literatúre je tento algoritmus uvádzaný v rôznych modifikáciách, ktoré sa zväčša líšia v konštrukcii počiatočného simplexu a vo voľbe zastavovacieho kritéria.

Nech  $S$  je nedegenerovaný simplex v  $\mathbb{R}^n$  s vrcholmi  $\mathbf{x}_0, \dots, \mathbf{x}_n$ . Označme  $f_j := f(\mathbf{x}_j)$ ,  $j = 0, \dots, n$  funkčné hodnoty v jeho vrcholoch. NM algoritmus pozostáva z nasledujúcich krokov:

#### 1. Generovanie počiatočného simplexu $S$

$S$  je zvyčajne generovaný  $n + 1$  vektormi  $\mathbf{x}_0, \dots, \mathbf{x}_n$  okolo vhodne zvoleného vstupného bodu  $\mathbf{x}_{in}$ . V praxi sa zvyčajne volí  $\mathbf{x}_0 = \mathbf{x}_{in}$ . Pre zvyšné  $n$  vrcholy volíme  $\mathbf{x}_j := \mathbf{x}_0 + h_j \mathbf{e}_j$ ,  $j = 1, \dots, n$ , kde  $\mathbf{e}_j$  je jednotkový vektor v smere  $j$ -tej súradnice a  $h_j \in \mathbb{R}$  je veľkosť kroku.

#### 2. Transformácia simplexu

Transformácia pozostáva z nasledujúcich krokov:

##### (a) Usporiadanie vrcholov

Vrcholy očísľujeme podľa ich funkčných hodnôt  $f_0 \leq f_1 \leq \dots \leq f_{n-1} \leq f_n$ . Z praktických dôvodov položíme  $b := 0$ ,  $m := n - 1$ ,  $w := n$ , kde tieto indexy označujú vrcholy podľa našej preferencie (best, medium, worst). Platí teda

$$f_w = \max_j f_j, \quad f_m = \max_{j \neq w} f_j, \quad f_b = \min_{j \neq w} f_j.$$

Ťažisko na stene oproti vrcholu  $\mathbf{x}_w$  je

$$\mathbf{c} := \frac{1}{n} \sum_{j \neq w} \mathbf{x}_j.$$

##### (b) Transformácia

Vytvoríme nový simplex zo súčasného simplexu nahradením vrcholu  $\mathbf{x}_w$  vrcholom  $\mathbf{x}_b^*$ . K transformácii použijeme štyri voliteľné parametre  $\alpha > 0$ ,  $0 < \beta < 1$ ,  $\gamma > 1$ ,  $\gamma > \alpha$ ,  $0 < \delta < 1$ . Spočítame  $\mathbf{x}_r$ , obraz bodu  $\mathbf{x}_w$ :  $\mathbf{x}_r := \mathbf{c} + \alpha(\mathbf{c} - \mathbf{x}_w)$ . Označme  $f_r := f(\mathbf{x}_r)$ . Nastáva jedna z nasledujúcich možností:

- Ak  $f_b \leq f_r \leq f_m$ , akceptujeme  $\mathbf{x}_b^* := \mathbf{x}_r$ .
- Ak  $f_r < f_b$ , spočítame ďalší vektor  $\mathbf{x}_e$ :  $\mathbf{x}_e := \mathbf{c} + \gamma(\mathbf{x}_r - \mathbf{c})$  a označíme  $f_e := f(\mathbf{x}_e)$ . Potom, keď  $f_e < f_r$ , akceptujeme  $\mathbf{x}_b^* := \mathbf{x}_e$ . Ak  $f_e \geq f_r$ , akceptujeme  $\mathbf{x}_b^* := \mathbf{x}_r$ .
- Ak  $f_m \leq f_r < f_w$ , tak spočítame  $\mathbf{x}_c := \mathbf{c} + \beta(\mathbf{x}_r - \mathbf{c})$ , označíme  $f_c := f(\mathbf{x}_c)$ . Ak  $f_c \leq f_r$ , akceptujeme  $\mathbf{x}_b^* := \mathbf{x}_c$  a ukončíme iteráciu, inak zrátame nový simplex  $S^*$

- Ak  $f_r \geq f_w$ , tak spočítame  $\mathbf{x}_c := \mathbf{c} + \beta(\mathbf{x}_w - \mathbf{c})$  a označíme  $f_c := f(\mathbf{x}_c)$ . Ak  $f_c < f_w$ , tak akceptujeme  $\mathbf{x}_b^* := \mathbf{x}_c$  a ukončíme iteráciu, inak zrátame nový simplex  $S^*$ .

Simplex  $S^*$  definujeme vrcholmi  $\mathbf{x}_j^* := \mathbf{x}_b + \delta(\mathbf{x}_j - \mathbf{x}_b)$ ,  $j = 1, \dots, n$ .

### 3. Test ukončenia iterácií

Nech  $S^k$  je simplex v  $k$ -tej iterácii NM algoritmu s vrcholmi  $\mathbf{x}_0^k, \dots, \mathbf{x}_n^k$ . Označme  $\mathbf{x}_b^k$  a  $\mathbf{x}_w^k$  vrcholy simplexu  $S^k$  s najmenšou a najväčšou funkčnou hodnotou. Jedným z možných kritérií na ukončenie iterácií je, ak vzdialenosti a funkčné hodnoty najlepších vrcholov následných simplexov v iteráciách sa od seba málo líšia:

$$|f(\mathbf{x}_b^k) - f(\mathbf{x}_b^{k-1})| < \varepsilon \quad \text{a} \quad \|\mathbf{x}_b^k - \mathbf{x}_b^{k-1}\| < \varepsilon,$$

kde  $\varepsilon > 0$  je dostatočne malé. S týmto kritériom pracuje napríklad software Mathematica. Ďalšou možnosťou je napríklad kritérium malých rozdielov funkčných hodnôt na vrchole simplexu:

$$|f(\mathbf{x}_b^k) - f(\mathbf{x}_w^k)| < \varepsilon,$$

pre  $\varepsilon > 0$  dostatočne malé. Prirodzeným kritériom je maximálny dovolený počet iterácií, napríklad v prípade, keď algoritmus nedosiahne vyššie zmienené kritéria.

Táto metóda nie je samozrejme skutočným globálnym optimalizátorom. Pre problémy s komplikovaným tvarom účelovej funkcie (napríklad nekonvexnosť) je tento algoritmus nevhodné použiť z obavy o uviaznutí na lokálnom minime. Pre niektoré úlohy však má tendenciu vyhnúť sa lokálnemu minimu. Je vhodná pre dobre podmienené problémy (striktná konvexita účelovej funkcie a podobne). V týchto prípadoch je rýchlejšia ako niektoré priame metódy, čo je jej výhoda. V prvých krokoch iterácie tejto metódy dochádza k výrazným zlepšeniam v hodnote účelovej funkcie. Je preto vhodné ju použiť v problémoch, u ktorých práve nepotrebujeme presné minimum, ale len určité zlepšenie v riešení. Niekoľko rokov numerického testovania tejto metódy ukázalo, že možnosť, keď musíme počítať všetky vrcholy nového simplexu  $S^*$ , sa v priebehu algoritmu vyskytuje výnimočne. To znamená, že pri iterácii nového simplexu potrebujeme zrátať jeden alebo dva testovacie body a ich funkčnú hodnotu. Vďaka tejto skutočnosti, kroky v algoritme prebiehajú veľmi rýchlo a efektívne aj keď minimalizujeme funkciu náročnú na jej funkčné ohodnotenie. Pri zahrnutí ohraničení v probléme je v tomto algoritme vhodné pretransformovať problém na neohraničený, pomocou vhodnej penalizačnej funkcie.

## 3.2 Simulované tavenie (SA)

Simulované tavenie (Simulated Annealing) je stochastický, alebo randomizovaný globálny optimalizačný algoritmus, inšpirovaný procesom tavenia a kontrolovaného chladenia kovu, s cieľom dosiahnuť minimálnu energiu kryštalickej štruktúry látky. SA algoritmus bol vyvinutý v roku 1983 k riešeniu kombinatorických a vysoko nelineárnych úloh. Výhodou SA je to, že svojimi vlastnosťami umožňuje nájsť globálne optimum. Názov a inšpirácia tejto metódy pochádzajú z odboru metalurgie. Žhavenie spôsobuje, že atómy v štruktúre látky sa stávajú nestabilnými a náhodne sa pohybujú po okolí pôvodnej polohy. Kontrolované chladenie umožňuje atómom nájsť polohu (konfiguráciu) s menšou internou energiou ako

interná energia atómov v počiatočnej polohe, a tak dosiahnúť lepšie vlastnosti látky. V analógii s týmto fyzikálnym procesom, každý krok SA algoritmu nahrádza súčasné riešenie novým, (kvázi)náhodným riešením, vybraným s pravdepodobnosťou, ktorá závisí na zmene hodnoty účelovej funkcie a na globálnom parametri  $T$ . V súvislosti s touto analógiou, parameter  $T$  nazývame *systémová teplota*, ktorý v priebehu algoritmu postupne znižujeme. Nové riešenia generujeme takmer náhodne ak je teplota  $T$  vysoká, s kladnou pravdepodobnosťou pripúšťame aj riešenia u ktorých vzrastie hodnota účelovej funkcie. Postupne so znižovaním  $T$  znižujeme aj pravdepodobnosť generovania horších riešení. Kladná pravdepodobnosť generovania horšieho riešenia zabraňuje tomu, aby algoritmus uviazol na lokálnom minime. Metóda bola nezávisle predstavená S. Kirkpatrickom, C. D. Gelattom a M. P. Vecchim v roku 1983 a v roku 1985 V. Černým, ako adaptácia Metropolis-Hastingovho algoritmu a metódy Monte Carlo generujúcej stavy termodynamických systémov.

### 3.2.1 Základný algoritmus

Hlavnou výhodou SA algoritmu je schopnosť sa vyhnúť lokálnemu minimu. Algoritmus náhodne hľadá nové riešenie, pričom s kladnou pravdepodobnosťou akceptuje aj tie, u ktorých hodnota účelovej funkcia vzrastie. Nech  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  je účelová funkcia, ktorú minimalizujeme. Formálne popíšeme štruktúru všeobecného SA algoritmu:

- **krok 0:** Nech  $\mathbf{x}_0$  je štartovací bod (riešenie). Položme  $Z_0 := \{\mathbf{x}_0\}$  a  $k := 0$ .
- **krok 1:** Generujme skúšobné riešenie  $\mathbf{y}_{k+1}$  z rozdelenia potencionálnych riešení  $D(Z_k)$ .
- **krok 2:** Generujme číslo  $p$  z rozdelenia  $R[0, 1]$  a položme

$$\mathbf{x}_{k+1} = \begin{cases} \mathbf{y}_{k+1} & \text{ak } p < A(\mathbf{x}_k, \mathbf{y}_{k+1}, T_k) \\ \mathbf{x}_k & \text{inak,} \end{cases} \quad (3.1)$$

kde  $A$  nazývame *akceptačnú funkciu (acceptance function)* s hodnotami v intervale  $[0, 1]$  a  $T_k$  *systémovú teplotu* v  $k$ -tej iterácii.

- **krok 3:** Nastavme  $Z_{k+1} = Z_k \cup \{\mathbf{y}_{k+1}\}$ . Množina  $Z_k$  obsahuje všetky informácie o iterovaných riešeniach do bodu  $k$ .
- **krok 4:** Nastavme  $T_{k+1} = U(Z_{k+1})$ , kde  $U$  je nezáporná funkcia, nazvaná *schéma chladenia (cooling schedule)*.
- **krok 5:** Skontrolujeme *ukončovacie pravidlo (stopping criterion)*. Keď vyhovuje ukončíme algoritmus, inak nastavíme  $k := k + 1$  a vrátime sa na krok 1.

V popise algoritmu sme bližšie nešpecifikovali jeho komponenty  $A$ ,  $D$ ,  $T_k$  a ukončovacie pravidlo. Voľba týchto komponentov si vyžaduje zvlášť veľkú pozornosť. V literatúre sa modifikácie SA algoritmu líšia práve v ich voľbe a riadení. Jedná sa často o zložité postupy, ktoré sú zväčša súčasťou komerčných softwérov. Následne uvedieme a stručne popíšeme niektoré z najčastejších prístupov.

## Akceptačná funkcia $A$

Vo väčšine prípadov sa pre túto funkciu volí takzvaná *metropolitná funkcia*

$$A(\mathbf{x}, \mathbf{y}, T) = \min\left\{1, e^{-\frac{f(\mathbf{y})-f(\mathbf{x})}{T}}\right\}. \quad (3.2)$$

Metropolitná funkcia vždy akceptuje kroky v ktorých hodnota účelovej funkcie u kandidáta  $\mathbf{y}_{k+1}$  poklesne vzhľadom k jej hodnote v bode  $\mathbf{x}_k$ . V opačnom prípade ich akceptuje práve s pravdepodobnosťou  $A(\mathbf{x}_k, \mathbf{y}_{k+1}, T_k)$ , kvôli tomu, aby sme neuviazli na lokálnom minime. Táto pravdepodobnosť je kontrolovaná pomocou systémovej teploty  $T_k$ . Ako sme už poznamenali, parameter  $T_k$  postupne znižujeme, a tým znižujeme aj túto pravdepodobnosť akceptovania „horších“ krokov.

Dalšia možnosť ako definovať túto funkciu je takzvané *Barkerovo kritérium*:

$$A(\mathbf{x}, \mathbf{y}, T) = \min\left\{\frac{1}{1 + e^{\frac{f(\mathbf{y})-f(\mathbf{x})}{T}}}\right\}. \quad (3.3)$$

Toto kritérium dokonca nemusí akceptovať kroky u ktorých účelová funkcia poklesne, ale so znižujúcou teplotou tieto kroky neakceptuje s vysokou pravdepodobnosťou.

Tieto dve akceptačné funkcie sú základné a ukázalo sa, že iné funkcie, ktoré majú niektoré vlastnosti zhodné sú určitým spôsobom ekvivalentné jednej z tejto dvojice funkcií. Detaily napríklad v [8].

## Schéma chladenia a generovanie nových riešení

Voľba schémy chladenia  $U$  a generovanie nového náhodného riešenia s rozdelením  $D$  je najdôležitejšou súčasťou definovania SA algoritmu. Uvedieme tu niektoré možnosti najčastejšie sa vyskytujúce v literatúre. Pre schému chladenia sa bežne volí

$$U(Z_k) = \beta(f(\mathbf{x}_k) - f^*)^g, \quad (3.4)$$

kde  $\beta, g > 0$  sú konštanty a  $f^*$  je hodnota účelovej funkcie v optimálnom bode. Samozrejme, hodnota  $f^*$  je vo väčšine prípadoch neznáma, preto radšej implementujeme jej odhad  $\hat{f}$ . Voľba parametrov  $\beta$  a  $g$  často závisí na niektorých vlastnostiach funkcie  $f$ , ktoré niektoré modifikácie SA odhadujú v priebehu algoritmu.

Rozloženie nového generovaného riešenia môžeme všeobecne napísať v tvare

$$\mathbf{y}_{k+1} = \mathbf{x}_k + \Delta r \boldsymbol{\theta}_k, \quad (3.5)$$

kde  $\boldsymbol{\theta}_k$  je náhodný vektor spĺňajúci  $\|\boldsymbol{\theta}_k\| = 1$  a  $\Delta r$  je veľkosť kroku. Generovanie  $\mathbf{y}_{k+1}$  v okolí  $\mathbf{x}_k$  v každom smere s rovnakou pravdepodobnosťou prináša určité problémy. Preto niekedy využívame aj informáciu o lokálnej štruktúre funkcie  $f$  a preferujeme v kroky v istých smeroch

$$\mathbf{y}_{k+1} = \mathbf{x}_k + \mathbf{Q}\mathbf{u}, \quad (3.6)$$

kde  $\mathbf{Q}$  je matica ktorá nesie informáciu o lokálnej štruktúre  $f$  a  $\mathbf{u}$  je  $n$ -rozmerný náhodný vektor s rozdelením  $R[-\sqrt{3}, \sqrt{3}]^n$ . Matica  $\mathbf{Q}$  by mala byť upravovaná v priebehu algoritmu, aby niesla aktuálne informácie o lokálnej štruktúre.



V takzvanom *adaptívnom algoritme simulovaného tavenia (Adaptive Simulated Annealing)*, rozdelenie nového riešenia závisí taktiež na systémovej teplote. V tomto prípade pre hustotu rozloženia kroku volíme

$$g_k(\Delta \mathbf{x}) = (2\pi T_k)^{-\frac{n}{2}} e^{-\frac{\|\Delta \mathbf{x}\|^2}{2T_k}}. \quad (3.7)$$

Pre takzvaný *Fast Annealing* algoritmus je spomínaná hustota

$$g_k(\Delta \mathbf{x}) = \frac{T_k}{(\|\Delta \mathbf{x}\|^2 + T_k^2)^{\frac{n+1}{2}}}.$$

Je nutné poznamenať, že voľby týchto postupov si vyžadujú upraviť schému chladenia pre konkrétnu modifikáciu. Podrobnejší popis čitateľ nájde napríklad v [8].

### Ukončovacie pravidlo

Vzhľadom k charakteru tohoto algoritmu je ťažké, ak nie nemožné určiť, či proces iterácií dosiahol globálneho minima (alebo jeho priblíženie s danou presnosťou). Doporučuje sa skončiť, ak po určitom počte cykloch (20-50) neakceptujeme nové riešenie. Prirodzeným pravidlom je, ak iterované riešenia sa začínú od seba málo líšiť:

$$|f_i - f_{i-u}| \leq \varepsilon, \quad u = 1, \dots, N_\varepsilon,$$

pre nejaké malé  $\varepsilon > 0$  a celé číslo  $N_\varepsilon$ . Systém Mathematica aplikuje iterácie tohto algoritmu pre viac štartovacích riešení (ich počet možno meniť). Ak označíme symbolom  $\mathbf{x}_k^b$  riešenie s najmenšou funkčnou hodnotou získané do  $k$ -tej iterácie, tak zastavovacie pravidlo aplikované týmto systémom je:

$$|f(\mathbf{x}_k^b) - f(\mathbf{x}_{k-1}^b)| < \varepsilon \quad \text{a} \quad \|\mathbf{x}_k^b - \mathbf{x}_{k-1}^b\| < \varepsilon,$$

kde  $\varepsilon > 0$  je dostatočne malé číslo.

Existujú dôkazy pre niektoré špecifické úlohy pri určitých podmienkach o konvergencii (s určitou pravdepodobnosťou) tejto metódy (detaily napríklad v [8]). V týchto prípadoch je potrebné systémovú teplotu znižovať pomaly, potom ale algoritmus je neefektívny. Metóda nemá veľké nároky na funkčné ohodnotenie účelovej funkcie pri iterácii kroku, takže je ju vhodné použiť pre problémy s účelovou funkciou náročnou na ohodnotenie a je efektívna pri optimalizovaní v diskretnom priestore (napríklad problém cestujúceho obchodníka). Algoritmus možno tiež implementovať paralelne, kde je úloha riešená súčasne na viacerých počítačoch.

## 3.3 Metóda diferenciálnych evolúcií (DE)

Metóda diferenciálnych evolúcií, alebo Differential Evolution je stochastický, populačne založený globálny optimalizačný algoritmus. Jeho autormi sú Kenneth Price a Reiner Storm, vyvinuli ho v rokoch 1994-1996 a od tej doby bolo predstavených mnoho modifikácií.

Princíp DE spočíva v tom, že na začiatku uvažujeme istú množinu riešení (populáciu), z ktorej v priebehu algoritmu vytvárame postupne nové množiny riešení (generácie potomkov). V analógii s biologickou evolúciou, generácie vytvárame tak, aby dedili najlepšie

vlastnosti. V našom prípade minimalizačného problému ide o dedenie funkčných hodnôt členov jednotlivých generácií. V nasledujúcej časti popíšeme základný princíp fungovania DE algoritmu.

### 3.3.1 Základný algoritmus

Algoritmus vytvára kandidátov do novej generácie riešení kombinovaním takzvaných rodičovských a niekoľkých ďalších riešení zo súčasnej generácie. Kandidát (riešenie) nahradí rodičovské riešenie vtedy, ak je funkčná hodnota účelovej funkcie kandidáta menšia ako u rodičovského riešenia. Algoritmus má štyri kľúčové parametre: *mutačný faktor* (*mutation factor*)  $F$ , *prechodovú pravdepodobnosť* (*crossover probability*)  $CR$  a *veľkosť populácie* ( $NP$ ). V pôvodnom DE algoritme sú tieto parametre pevné, ale časom vznikli postupy ako ich v priebehu meniť a dosiahnuť lepších výsledkov. Úlohu týchto parametrov objasníme neskôr.

Algoritmus prebieha v štyroch fázach: *inicializácia* (*initialization*), *mutácia* (*mutation*), *rekombinácia* (*recombination*) a *selekcia* (*selection*). Predpokladajme, že hľadáme globálne minimum funkcie  $f : X \rightarrow \mathbb{R}$ ,  $X \subset \mathbb{R}^n$ . Označme generácie riešení pozostávajúce z  $NP$  vektorov

$$\mathbf{x}_{i,G} = (x_{1,i,G}, \dots, x_{n,i,G}), \quad i = 1, \dots, NP,$$

kde  $G$  je číslo generácie.

#### Inicializácia

Počiatočnú generáciu riešení volíme náhodne, pričom jej jedinci splňajú

$$x_j^L \leq x_{j,i,1} \leq x_j^U, \quad i = 1, \dots, NP, \quad j = 1, \dots, n,$$

kde  $x_j^L$  a  $x_j^U$  sú vhodne zvolené čísla z  $\mathbb{R}$ . Hranice volíme v závislosti na úlohe. Následne každý vektor generácie prejde fázou mutácie, rekombinácie a operácia selekcie vyberie jedinca do ďalšej generácie  $G + 1$ .

#### Mutácia

Mutácia vytvorí pre každého rodičovského jedinca  $\mathbf{x}_{i,G}$ ,  $i = 1, \dots, NP$  generácie  $G$  *mutačný vektor* (niekedy označovaný ako donor vektor)  $\mathbf{v}_{i,G}$ . Mutačný vektor vytvoríme užitím jednej z viacerých mutačných stratégií. Niektoré z najčastejšie používaných sú:

- **rand/1**:  $\mathbf{v}_{i,G+1} = \mathbf{x}_{r1,G} + F(\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G})$
- **best/1**:  $\mathbf{v}_{i,G+1} = \mathbf{x}_{best,G} + F(\mathbf{x}_{r1,G} - \mathbf{x}_{r2,G})$
- **rand/2**:  $\mathbf{v}_{i,G+1} = \mathbf{x}_{r1,G} + F(\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G}) + F(\mathbf{x}_{r4,G} - \mathbf{x}_{r5,G})$
- **best/2**:  $\mathbf{v}_{i,G+1} = \mathbf{x}_{best,G} + F(\mathbf{x}_{r2,G} - \mathbf{x}_{r3,G}) + F(\mathbf{x}_{r4,G} - \mathbf{x}_{r5,G})$

kde  $r1, r2, r3, r4, r5$  sú náhodne, navzájom rôzne zvolení jedinci z množiny  $[1, NP] \setminus i$ .  $F \in [0, 2]$  je konštanta nazývaná *mutačný faktor*, ale zvyčajne je menšia ako 1 a  $\mathbf{x}_{best,G}$  značí najlepší jedinca v generácii  $G$ , teda riešenie s najmenšou funkčnou hodnotou účelovej funkcie  $f$ .

## Rekombinácia

Pomocou  $\mathbf{x}_{i,G}$  a mutačného vektora  $\mathbf{v}_{i,G}$  vytvoríme *skúšobný vektor*  $\mathbf{u}_{i,G}$  (kandidáta), ktorý je následne porovnaný s rodičovským vektorom  $\mathbf{x}_{i,G}$  a jeden z nich je vybraný do ďalšej generácie.

$$\mathbf{u}_{j,i,G} = \begin{cases} \mathbf{v}_{j,i,G} & \text{ak } rand_{j,i} \leq CR \text{ alebo } j = j_{rand} \\ \mathbf{x}_{j,i,G} & \text{ak } rand_{j,i} > CR \text{ a } j \neq j_{rand}, \end{cases} \quad (3.8)$$

kde  $rand_{j,i}$  je náhodne generované číslo z rozdelenia  $R[0,1]$  a  $j_{rand}$ , je číslo z intervalu  $[1, \dots, NP]$ .  $j_{rand}$  zaisťuje to, že  $\mathbf{v}_{i,G} \neq \mathbf{x}_{i,G}$ . CR je zvolená konštanta z intervalu  $[0,1]$  nazývaná *prechodová pravdepodobnosť (crossover probability)*.

## Selekcia

Operácia selekcie vyberie ďalšej generácii potomka nasledujúcim pravidlom:

$$\mathbf{x}_{i,G+1} = \begin{cases} \mathbf{u}_{i,G+1} & \text{ak } f(\mathbf{u}_{i,G+1}) \leq f(\mathbf{x}_{i,G}) \\ \mathbf{x}_{j,i,G} & \text{inak.} \end{cases} \quad (3.9)$$

Mutáciu, selekciu a rekombináciu opakujeme pokiaľ nedosiahneme nejaké zvolené kritérium. Môže to byť napríklad dosiahnutý určitý počet iterácií, alebo ak sa funkčné hodnoty vektorov generácie, poprípade ich vzdialenosti málo líšia. Označme  $\mathbf{x}_b^k$  riešenie s najmenšou funkčnou hodnotou získané do generovania  $k$ -tej generácie. Iterácie môžeme teda ukončiť, ak napríklad nastane:

$$|f(\mathbf{x}_b^k) - f(\mathbf{x}_b^{k-1})| \leq \varepsilon \quad \text{a} \quad \|\mathbf{x}_b^k - \mathbf{x}_b^{k-1}\| \leq \varepsilon,$$

kde  $\varepsilon > 0$  je dostatočne malé zvolené číslo. S týmto kritériom pracuje systém Mathematica.

Dôležitou súčasťou algoritmu je generovanie nových riešení, ak je úloha ohraničená. Súčasne s vývojom tohto algoritmu vznikali aj postupy ako pracovať s ohraničeniami. Implementácia týchto postupov si však nevyžaduje vážnejšie zásahy do základného algoritmu. Jednou z možností je transformovať ohraničený problém na neohraničený použitím vhodnej penalizačnej funkcie. Metóda diferenciálnych evolúcií sa ukázala ako silný a robustný optimalizačný nástroj s uplatnením pre širokú škálu optimalizačných úloh. Jej modifikácie sú obľúbené a vo svete často používané. Pri testoch optimalizačných metód na niektorých testovacích funkciách dosiahla lepších výsledkov ako metóda simulovaného tavenia (pozri napríklad [13]). Metóda je náročná na počítanie funkčnej hodnoty účelovej funkcie. Možno ju však implementovať paralelným algoritmom, kde je hlavný algoritmus rozdelený na viacej subalgoritmov a tie sú iterované súbežne na viacerých počítačoch, čo rapídne zefektívňuje optimalizačný proces. V súčasnosti by bolo možné realizovať paralelne výpočty v prostredí systému Mathematica 7, ktorý paralelne výpočty ovláda programovo.

# Kapitola 4

## Aplikácie vo financiách a numerické ilustrácie

### 4.1 Problém optimálneho portfólia a finančné umiestnenie aktív

Problém voľby optimálneho portfólia je jedna z najznámejších aplikácií optimalizačných metód vo financiách. Teória voľby optimálneho portfólia bola vypracovaná americkým ekonómom Harrym Markowitzom, prezentovaná v roku 1952, za ktorú v roku 1990 obdržal Nobelovu cenu za ekonómiu. Markowitz navrhol dve kritéria pre voľbu portfólia, a to očakávaný výnos a riziko, pričom snahou je maximalizovať očakávaný výnos a minimalizovať riziko. Za riziko portfólia označil jednoducho jeho smerodajnú odchýlku očakávaného výnosu predstavujúceho náhodnú veličinu.

Časom však k tomuto prístupu vznikali rôzne alternatívy. Kritika tohto modelu sa týkala hlavne smerodajnej odchýlky ako voľby rizika, ktorá penalizuje záporný aj kladný výnos. Naproti tomu sa Markowitzov prístup stal základom pre ostatné modely, ktoré sa rovnako snažia maximalizovať očakávaný výnos a minimalizovať riziko, s použitím iných definícií rizika. Základný Markowitzov model tu popíšeme stručne, detailnejšie predpoklady a teóriu Markowitzovho portfólia čitateľ nájde napríklad v Dupačová a kol. [3].

Model predpokladá racionálneho investora, ktorý disponuje určitou sumou peňazí a rozhoduje sa, akými podielmi ich investovať do  $n$  cenných papierov, pričom miery výnosov jednotlivých aktív sú náhodné veličiny  $\rho_i, j = 1, \dots, n$  so známou strednou hodnotou (očakávaným výnosom)  $\mu_i = \mathbb{E}(\rho_i)$  a známym rozptylom  $\sigma_i^2 = \text{Var}(\rho_i)$ . Model ďalej predpokladá známu kovarianciu  $\sigma_{ij} = \text{cov}(\rho_i, \rho_j)$  pre každé dva cenné papiere  $i$  a  $j$ . Označme pomernú časť peňazí investovaných do cenného papiera  $i$  označením  $x_i$ , pre celé takto zvolené portfólio označme  $\mathbf{x} = (x_1, \dots, x_n)^\top$  a ďalej označme  $\boldsymbol{\rho} = (\rho_1, \dots, \rho_n)^\top$ ,  $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^\top$ . Pre celé portfólio potom očakávaný zisk a variancia je:

$$\begin{aligned}\mathbb{E}(\boldsymbol{\rho}^\top \mathbf{x}) &= \boldsymbol{\mu}^\top \mathbf{x}, \\ \text{Var}(\boldsymbol{\rho}^\top \mathbf{x}) &= \mathbf{x}^\top \mathbf{Q} \mathbf{x},\end{aligned}$$

kde  $\mathbf{Q}$  je kovariančná matica,  $\mathbf{Q}_{i,j} = \sigma_{i,j}$ .

Našou úlohou je maximalizovať očakávaný výnos a minimalizovať riziko investície za podmienky  $\mathbf{1}^\top \mathbf{x} = 1$ , a prípadne aj ďalších obmedzujúcich podmienok (napr. legislatívne

obmedzenia, zakázanie krátkych predajov). Označme  $X \subset \mathbb{R}^n$  ako množinu týchto prípustných portfólií.

Pripomeňme ďalej pojem *efektívne portfólio*. Úloha výberu portfólia sa potýka s maximalizáciou očakávaného výnosu a minimalizáciou rizika, čo sú dva protichodné úlohy (všeobecne portfólio s vyšším očakávaným výnosom vykazuje taktiež väčšie riziko). Označme pre portfólio  $\mathbf{x}$  jeho očakávaný výnos  $R(\mathbf{x})$  a riziko  $\phi(\mathbf{x})$ . Budeme hovoriť, že portfólio  $\mathbf{x}^*$  je *efektívne portfólio*, ak neexistuje také portfólio  $\mathbf{x}$ , že platí:

$$(R(\mathbf{x}^*) < R(\mathbf{x}) \wedge \phi(\mathbf{x}^*) \geq \phi(\mathbf{x})) \vee (R(\mathbf{x}^*) = R(\mathbf{x}) \wedge \phi(\mathbf{x}^*) > \phi(\mathbf{x})).$$

Inými slovami, portfólio  $\mathbf{x}^*$  je efektívne, ak nenájdeme iné portfólio, ktoré má väčší alebo rovnaký očakávaný výnos a menšie riziko, alebo má rovnaké riziko a väčší očakávaný výnos. Pod rizikom v tomto kontexte nemyslíme len smerodajnú odhýlku, ale aj iné miery rizika ktoré spomenieme neskôr. Množinu efektívnych portfólií budeme nazývať *efektívnou množinou*.

Úlohu optimálneho portfólia možno riešiť viacerými spôsobmi, ktoré su ekvivalentné v zmysle nasledujúceho tvrdenia.

**Tvrdenie 2** *Uvažujme nasledujúce optimalizačné úlohy:*

$$\begin{aligned} \min_{\mathbf{x}} \phi(\mathbf{x}) - \mu R(\mathbf{x}), \quad \mathbf{x} \in X, \quad \mu \geq 0, \\ \min_{\mathbf{x}} \phi(\mathbf{x}), \quad R(\mathbf{x}) \geq \rho, \quad \mathbf{x} \in X, \quad \rho \in \mathbb{R} \\ \min_{\mathbf{x}} -R(\mathbf{x}), \quad \phi(\mathbf{x}) \leq \omega, \quad \mathbf{x} \in X, \quad \omega \in \mathbb{R}, \end{aligned}$$

kde  $\phi(\mathbf{x})$ ,  $R(\mathbf{x})$  sú reálne funkcie premennej  $\mathbf{x} \in X \subset \mathbb{R}^n$ . Nech ohraničenia  $R(\mathbf{x}) \geq \rho$ ,  $\phi(\mathbf{x}) \leq \omega$  majú vnútorný bod. Ak je  $\phi(\mathbf{x})$  konvexná,  $R(\mathbf{x})$  konkávna a množina  $X$  je konvexná, potom tieto tri optimalizačné úlohy sú ekvivalentné v tom zmysle, že obmieňaním parametrov  $\mu$ ,  $\rho$ ,  $\omega$ , tieto úlohy generujú rovnakú efektívnu množinu.

Dôkaz napríklad v Krokmal a kol. [6]. Jedna z možností zápisu problému optimálneho portfólia (ako ho definoval Markowitz) je minimalizovanie rizika pri požiadavke minimálneho očakávaného výnosu  $\mu_p$ :

$$\begin{aligned} \min_{\mathbf{x}} \mathbf{x}^\top \mathbf{Q} \mathbf{x} \\ \mathbf{1}^\top \mathbf{x} = 1 \\ \boldsymbol{\mu}^\top \mathbf{x} \geq \mu_p. \end{aligned} \tag{4.1}$$

V literatúre sa môžeme často stretnúť s označením „Mean-Variance Optimization (MVO)“ pre túto optimalizačnú úlohu. Tento problém v takej forme ako je uvedený vyššie možno riešiť explicitne užitím metódy Lagrangeových multiplikátorov, detaily napríklad v Dupáčová a kol. [3]. V praxi však takýmto spôsobom nie je možné modelovať reálny problém. Okrem toho, že povoľujeme investovať do krátkych pozícií, opomenuli sme aj ďalšie kritéria ako transakčné náklady, dane, obmedzenia pri rebalancovaní portfólia a ďalšie obmedzenia týkajúce sa podskupín (sektorov) daných uvažovaných aktív.

#### 4.1.1 Základný model a reálne ohraničenia

Predpokladajme, že investor obchoduje na trhu s dostupnými  $n$  cennými papiermi  $i = 1, \dots, n$ . V súčasnosti vlastní portfólio, označme ho vektorom  $\mathbf{x}^0 = (x_1^0, \dots, x_n^0)^\top$ , ktoré sa

snaží rebalancovať (optimalizovať) vzhľadom k dostupným informáciám na trhu a svojím preferenciám. Z praktických dôvodov, nech zložky vektora  $x_i^0$ ,  $i = 1, \dots, n$ , na rozdiel od vyššie uvedeného, označujú počet držaných jednotiek (podielov) aktíva  $i$ ,  $i = 1, \dots, n$  (napríklad počet zakúpených akcií). Podobne, označme vektorovou premennou  $\mathbf{x} = (x_1, \dots, x_n)^\top$  hľadané rebalancované portfólio. Zaveďme ďalej nasledujúce pomocné označenia:

$n$	počet dostupných cenných papierov
$G$	počet segmentov cenných papierov
$G_g$	množina aktív patriaca do segmentu $g$ , $g = 1, \dots, G$
$\varepsilon_i$	minimálna povolená časť kapitálu v $i$ -tom aktíve, $i = 1, \dots, n$
$\delta_i$	maximálna povolená časť kapitálu v $i$ -tom aktíve, $i = 1, \dots, n$
$u_i^s$	maximálny počet jednotiek aktíva $i$ na predaj, $i = 1, \dots, n$
$u_i^b$	dostupný počet jednotiek aktíva $i$ na nákup, $i = 1, \dots, n$
$\xi_j$	minimálna povolená časť kapitálu v $j$ -tom segmente, $j = 1, \dots, G$
$\vartheta_j$	maximálna povolená časť kapitálu v $j$ -tom segmente, $j = 1, \dots, G$
$c_i$	transakčné náklady na nákup/predaj aktíva $i$ , $i = 1, \dots, n$
$q_i$	súčasná tržná cena aktíva $i$ , $i = 1, \dots, n$
$\Theta$	maximálny povolený celkový obrat rebalancovania (nákupu/predaja)
$\mu_p$	požadovaná očakávaná miera výnosu rebalancovaného portfólia.

Následne popíšeme najbežnejšie ohraničenia s ktorými sa investor potýka na trhu.

### Transakčné náklady

Transakčné náklady, väčšinou vo forme provízie obchodníkovi, majú nezanedbateľný vplyv na výsledné optimálne portfólio. Uvažujme, ako je v praxi bežné, lineárne transakčné náklady. Investor za nákup alebo predaj inštrumentu  $i$  zaplatí časť  $c_i$  transakčnej čiastky. Zahrňme túto skutočnosť do ohraničení. Pre celú transakciu (rebalancovanie) teda platí:

$$\sum_{i=1}^n q_i x_i^0 = \sum_{i=1}^n c_i q_i |x_i^0 - x_i| + \sum_{i=1}^n q_i x_i. \quad (4.2)$$

Túto rovnosť upravíme ďalej použitím pomocných premenných  $u_i^+$  a  $u_i^-$ ,  $i = 1, \dots, n$  na lineárne ohraničenia:

$$\begin{aligned} \sum_{i=1}^n q_i x_i^0 &= \sum_{i=1}^n c_i q_i (u_i^+ + u_i^-) + \sum_{i=1}^n q_i x_i, \\ x_i - x_i^0 &= u_i^+ - u_i^-, \\ u_i^+ &\geq 0, u_i^- \geq 0, \quad i = 1, \dots, n. \end{aligned}$$

Premenné  $u_i^+$ ,  $u_i^-$ ,  $i = 1, \dots, n$  teda vyjadrujú zmeny pozícií, počet kúpených, respektíve predaných jednotiek aktíva  $i$ ,  $i = 1, \dots, n$ . Ak má byť hodnota celej transakcie pri rebalancovaní portfólia obmedzená hodnotou  $\Theta$ , tak zápisom máme:

$$\sum_{i=1}^n q_i |x_i - x_i^0| \leq \Theta, \quad (4.3)$$

a v prípade vyššie použitých lineárnych ohraňení to nie je nič iné ako:

$$\sum_{i=1}^n (u_i^+ + u_i^-) q_i \leq \Theta.$$

### Ohraničenia likvidity

Uvažujeme ďalej, že zmeny pozícií môžu byť ohraňené. Zápisom máme:

$$0 \leq u_i^- \leq u_i^s, \quad 0 \leq u_i^+ \leq u_i^b, \quad i = 1, \dots, n.$$

Toto ohraňenie môže predstavovať napríklad obmedzenú likviditu na trhu, alebo dostupný počet aktív na nákup.

### Ohraničenia hodnôt

Často sa pre jednotlivé aktíva zavádzajú ohraňenia v pomere k hodnote celého portfólia:

$$\begin{aligned} q_i x_i &\leq \delta_i \sum_{k=1}^n x_k q_k, \\ q_i x_i &\geq \varepsilon_i \sum_{k=1}^n x_k q_k, \quad i = 1, \dots, n. \end{aligned}$$

Podobne, môžeme tiež zaviesť ohraňenia pre pozície aktív v rôznych skupinách (segmentoch). Ohraňenia pre alokovaný kapitál v týchto segmentoch sú:

$$\begin{aligned} \sum_{j \in G_g} x_j q_j &\leq \vartheta_g \sum_{j=1}^n x_j q_j, \\ \sum_{j \in G_g} x_j q_j &\geq \xi_g \sum_{j=1}^n x_j q_j, \quad g = 1, \dots, G. \end{aligned}$$

S takými obmedzeniami sa väčšinou stretávajú penzijné fondy, poisťovne a iní investori, ktorých finančné umiestnenie aktív je do istej miery obmedzované zákonom. Tieto skupiny tak môžu prezentovať napríklad akcie od emitentov s určitým ratingovým hodnotením, alebo jedna skupina môže predstavovať napríklad štátne dlhopisy. Všeobecne povedané, v jednej skupine sú inštrumenty s rovnakou vlastnosťou z určitého hľadiska.

### Požadovaný výnos

V niektorých úlohách požadujeme od portfólia určitú minimálnu požadovanú mieru výnosu  $\mu_p$ . Tento požiadavok zavedieme do ohraňení nasledujúcou nerovnosťou:

$$\sum_{i=1}^n (\mu_i + 1) q_i x_i - \sum_{i=1}^n q_i x_i^0 \geq \mu_p \sum_{i=1}^n q_i x_i^0. \quad (4.4)$$

## 4.1.2 Alternatívne prístupy k problému optimálneho portfólia

### Prístup autorov Kono a Yamazaki

Smerodajná odchýlka v Markowitzovom modeli ako miera rizika, bola často kritizovaná kvôli niektorým nevhodným vlastnostiam. Fakt, že penalizuje záporný aj kladný výnos, vedie často k nevhodnej voľbe portfólia. Námietky voči symetrii rozptylu výnosu ako miery

rizika viedli k definícii asymetrickej miery rizika, *kvadratickej semivariancie*. Pre portfólio prezentované váhovým vektorom to je:

$$\mathbb{E}\left(\left[\sum_{i=1}^n \mu_i x_i - \sum_{i=1}^n \rho_i x_i\right]^+\right)^2.$$

Nevýhodou je nárast obtiažnosti numerického výpočtu riešeného optimalizačného problému. William Forsyth Sharpe navrhol definovať riziko ako *strednú absolútnu odchýlku* (*mean absolute deviation*):

$$\omega(x) := \mathbb{E}\left|\sum_{i=1}^n \mu_i x_i - \sum_{i=1}^n \rho_i x_i\right|.$$

Sharpeho myšlienku potom realizovali Kono a Yamazaki. Riešili úlohu nájsť portfólio efektívne vzhľadom k  $\boldsymbol{\mu}^\top \mathbf{x}$  a  $-\omega(\mathbf{x})$ . Za predpokladu, že náhodný vektor  $\boldsymbol{\mu}$  má  $n$ -rozmerné normálne rozdelenie, sú hodnoty  $\omega(\mathbf{x})$  násobkami smerodajnej odchýlky. Konkrétne  $\omega(\mathbf{x}) = \sqrt{2}\pi\sigma(\mathbf{x})$ , vid' napríklad Cornuejols a kol. [2]. Za tohto predpokladu, Markowitzov model môže byť formulovaný ako úloha:

$$\begin{aligned} \min_{\mathbf{x}} \mathbb{E}\left|\sum_{i=1}^n \mu_i x_i - \sum_{i=1}^n \rho_i x_i\right| \\ \sum_{i=1}^n \mu_i x_i \geq \mu_p. \end{aligned}$$

Výhodou tohto prístupu je, že nemusíme odhadovať variančnú maticu. Navyše, ak nahradíme stredné hodnoty výberovými priermi z historických dát, môžeme túto úlohu previesť na úlohu lineárneho programovania.

Nech vektory  $\mathbf{r}_t = (r_{1t}, \dots, r_{nt})$ ,  $t = 1, \dots, T$  sú realizáciou  $n$ -rozmerného náhodného vektoru  $\boldsymbol{\rho} = (\rho_1, \dots, \rho_n)$ , získané simuláciou užitím metódy Monte Carlo, projekciou z vhodného modelu, alebo získané z historických dát. Potom pre odhad strednej absolútnej odchýlky máme:

$$\begin{aligned} \mu_i &= \frac{1}{T} \sum_{t=1}^T r_{it} \\ \mathbb{E}\left|\sum_{i=1}^n (\rho_i - \mu_i) x_i\right| &= \frac{1}{T} \sum_{t=1}^T \left|\sum_{i=1}^n (r_{it} - \mu_i) x_i\right|. \end{aligned}$$

Kôli získaniu lineárneho tvaru, doplníme do druhej rovnosti pomocné premenné. Takže výsledný lineárny program má tvar:

$$\begin{aligned} \min \sum_{t=1}^T (y_t + z_t) \\ y_t - z_t &= \sum_{i=1}^n (r_{it} - \mu_i) x_i, \quad t = 1, \dots, T \\ \sum_{i=1}^n \mu_i x_i &\geq \mu_p \\ \sum_{i=1}^n x_i &= 1 \\ x_i &\geq 0, \quad i = 1, \dots, n \\ y_t &\geq 0, z_t \geq 0, \quad t = 1, \dots, T. \end{aligned} \tag{4.5}$$

Keďže máme lineárny program, môžeme ho efektívne riešiť i s veľkým počtom neznámych, napríklad použitím metódy vnútorného bodu pre lineárne programovanie. Označenia  $x_i$ ,  $i = 1, \dots, n$  tu opäť predstavujú váhy jednotlivých inštrumentov v portfóliu. Jednoduchou úpravou však možno získať model s počtami inštrumentov a tak ho rozšíriť o spomínané ohraňovania.



## Sharpeho koeficient

Maximalizovaním takzvaného Sharpeho koeficientu, alebo takzvanej Sharpeho miery portfólia

$$s(\mathbf{x}) = \frac{\boldsymbol{\mu}^\top \mathbf{x}}{\sqrt{\mathbf{x}^\top \mathbf{Q} \mathbf{x}}} \quad (4.6)$$

dostaneme takzvané *optimálne rizikové portfólio*. Je to bod dotyku *priamky kapitálového trhu (capital asset line)* a efektívnej množiny (Markowitzovho portfólia). Uvažujme optimalizačný problém maximalizovania Sharpeho koeficientu, pre úplnosť doplníme vo všeobecnej forme ďalšie ohraničenia.

$$\begin{aligned} \max \quad & \frac{\boldsymbol{\mu}^\top \mathbf{x}}{\sqrt{\mathbf{x}^\top \mathbf{Q} \mathbf{x}}} \\ \mathbf{A} \mathbf{x} = & \mathbf{b} \\ \mathbf{C} \mathbf{x} \geq & \mathbf{d}. \end{aligned} \quad (4.7)$$

V takejto forme je tento problém neľahko riešiť. Účelová funkcia nie je totiž všeobecne konkávná. Túto úlohu však možno previesť na ekvivalentnú minimalizačnú úlohu s konvexnou účelovou funkciou. Označme množiny

$$\begin{aligned} \chi &:= \{\mathbf{x} \in \mathbb{R}^n : \mathbf{A} \mathbf{x} = \mathbf{b}, \mathbf{C} \mathbf{x} \geq \mathbf{d}\}, \\ \chi^+ &:= \{\mathbf{x} \in \mathbb{R}^n, \kappa \in \mathbb{R} : \kappa > 0, \mathbf{x}/\kappa \in \chi\} \subset \mathbb{R}^{n+1}. \end{aligned}$$

Portfólio s maximálnym Sharpeho koeficientom nájdeme pomocou riešenia optimalizačného problému:

$$\{\min \sqrt{\mathbf{x}^\top \mathbf{Q} \mathbf{x}} : (\mathbf{x}, \kappa) \in \chi^+, \boldsymbol{\mu}^\top \mathbf{x} = \delta\}$$

kde  $\delta$  je vhodne zvolené číslo. Ak  $(\hat{\mathbf{x}}, \hat{\kappa})$  je optimálne riešenie tohto problému, tak portfólio s maximálnym Sharpeho koeficientom je  $\mathbf{x}^* = \hat{\mathbf{x}}/\hat{\kappa}$ . Toto je dôsledok faktu, že funkcia  $s(\mathbf{x})$  je homogénna vzhľadom k  $\mathbf{x}$  a množina  $\chi^+$  je kužeľ. Voľba parametra  $\delta$  je kvôli normalizácii čitateľa v  $s(\mathbf{x})$ . Volíme ho ľubovoľne tak, aby pre nejaké  $\mathbf{x}$  platilo  $\boldsymbol{\mu}^\top \mathbf{x} = \delta$ .

### 4.1.3 VaR a CVaR

*Value-at-Risk*, alebo *maximálna očakávaná strata* je široko používaná a akceptovaná miera finančného rizika naprieč spektrom priemyselných odvetví a účastníkmi trhu. Popularitu si získala aj vďaka jednoduchšej interpretácii a porozumeniu vysokých strát portfólia. Podľa definície, vzhľadom k parametru (hladine)  $\alpha$ , je  $\alpha$ -VaR portfólia najmenšia čiastka taká, že s pravdepodobnosťou  $\alpha$  ju strata nepresiahne. Slovanami povedané, VaR je kvantil rozdelenia straty (záporneho výnosu) portfólia, za predpokladu, že výnos je náhodná veličina. V tejto súvislosti, pre špecifikáciu kvantilu užívame označenie  $\alpha$ -VaR pre  $\alpha$ -kvantil rozdelenia straty. Napríklad 0.95-VaR je horný odhad straty, ktorú s pravdepodobnosťou 0.95 neprekročíme. Za predpokladu (log)normálnych rozdelení, dokážeme s rizikom VaR efektívne pracovať. Za predpokladu iných rozdelení, VaR ako miera rizika nadobúda nevhodných vlastností. Napríklad, nejedná sa o sub-aditívnu mieru rizika. VaR portfólia dvoch aktív môže byť väčšie ako súčet VaR dvoch samostatných aktív, diverzifikácia portfólia teda nemusí viesť k zníženiu VaR. Taktiež za predpokladu diskrétného rozdelenia, keď VaR počítame pomocou generovania scénarov (metódou Monte Carlo), VaR nie je konvexnou funkciou (ako funkcia pozícií v portfóliu) a nadobúda značného počtu lokálnych extrémov. Modelovanie rizika VaR rôznymi prístupmi môže čitateľ nájsť napríklad na URL

adrese <http://www.gloriamundi.org/>, alebo v technickom dokumente RiskMetrics [9]. Za predpokladu (log)normálnych rozdelení parametrov podkladových aktív sa najčastejšie používa prístup lineárnej aproximácie funkcie rizika VaR. Taktiež použitím historických dát a metódy Monte Carlo v prípade nelineárnych podkladových aktív, finančných derivátov, napríklad opcií.

Ďalšou, podobnou mierou rizika je *Conditional Value-at-Risk* (CVaR, Expected Shortfall, Mean Excess Loss, Mean Shortfall, Tail Value-at-Risk). Pre spojité rozdelenie je  $\alpha$ -CVaR portfólia definovaná ako podmienená očakávaná strata, za podmienky, že je väčšia alebo rovná, ako  $\alpha$ -VaR. CVaR je podobná miera rizika ako VaR, všeobecne má však atraktívnejšie vlastnosti ako VaR. CVaR je sub-aditívna a konvexná. Navyše bolo dokázané, že CVaR je koherentná miera rizika (pozri napríklad Pflug [11]). Numerické experimenty ukázali, že minimalizovanie CVaR vedie k podobným výsledkom (portfóliam) ako minimalizovanie VaR, pretože hodnota VaR je vzhľadom k definícii vždy menšia, alebo nanajvýš rovná hodnote CVaR. Navyše, ak je rozdelenie výnosu normálne, tieto dve miery sú ekvivalentné (viď [12]). Na druhej strane, pre „šikmé“ rozdelenia tieto dva prístupy môžu viesť k celkom odlišným výsledkom. Je viac prístupov ako tieto miery implementovať v probléme optimálneho portfólia. Na jednej strane môžeme minimalizovať riziko, na druhej strane obmedziť riziko jeho zavedením do ohraničenia. Vďaka atraktívnym vlastnostiam miery CVaR, linearizujeme problém optimálneho portfólia maximalizovania očakávaného výnosu s obmedzením miery rizika CVaR a následnému získaniu VaR (Krokhmal a Uryasev [6]). Na linearizovanie problému použijeme nasledujúci teoretický aparát.

Označme  $f(\mathbf{x}, \mathbf{y})$  ako stratu hodnoty portfólia v súvislosti s rozhodovacím vektorom (voľbou portfólia)  $\mathbf{x} \in X \subset \mathbb{R}^n$  (množina prípustných portfólií) a náhodným vektorom  $\mathbf{y} \in \mathbb{R}^n$  (napríklad neznáme budúce výnosy atd.). Nech pre každé  $\mathbf{x} \in X$  je  $f(\mathbf{x}, \mathbf{y})$  náhodná veličina, a nech náhodný vektor  $\mathbf{y}$  má hustotu  $p(\mathbf{y})$ . Označme

$$\Psi(\mathbf{x}, \zeta) = \int_{f(\mathbf{x}, \mathbf{y}) \leq \zeta} p(\mathbf{y}) d\mathbf{y}.$$

Pre jednoduchosť predpokladajme, že  $\Psi(\mathbf{x}, \zeta)$  je pre každé  $\mathbf{x} \in X$  spojitá v  $\zeta$ . Zavedme označenie  $\zeta_\alpha(\mathbf{x})$  pre  $\alpha$ -VaR a  $\phi_\alpha(\mathbf{x})$  pre  $\alpha$ -CVaR vzhľadom k rozhodovaciemu vektoru  $\mathbf{x}$ . Platí teda

$$\zeta_\alpha(\mathbf{x}) = \min\{\zeta \in \mathbb{R} : \Psi(\mathbf{x}, \zeta) \geq \alpha\} \quad (4.8)$$

a

$$\phi_\alpha(\mathbf{x}) = (1 - \alpha)^{-1} \int_{f(\mathbf{x}, \mathbf{y}) \geq \zeta_\alpha(\mathbf{x})} f(\mathbf{x}, \mathbf{y}) p(\mathbf{y}) d\mathbf{y}. \quad (4.9)$$

Definujme ďalej funkciu, ktorá ma v charakterizácii  $\phi_\alpha(\mathbf{x})$  a  $\zeta_\alpha(\mathbf{x})$  kľúčovú úlohu:

$$F_\alpha(\mathbf{x}, \zeta) = \zeta + (1 - \alpha)^{-1} \int_{\mathbf{y} \in \mathbb{R}^n} [f(\mathbf{x}, \mathbf{y}) - \zeta]^+ p(\mathbf{y}) d\mathbf{y}.$$

Za vyššie uvedených predpokladov má táto funkcia vhodné vlastnosti k určaniu optimálneho portfólia.

**Tvrdenie 3** *Pre každé  $\mathbf{x} \in \mathbb{R}^n$  je funkcia  $F_\alpha(\mathbf{x}, \zeta)$  premennej  $\zeta$  konvexná, spojitá diferencovateľná a platí:*

$$\phi_\alpha(\mathbf{x}) = \min_{\zeta \in \mathbb{R}} F_\alpha(\mathbf{x}, \zeta).$$

Množina hodnôt  $\zeta$  pre ktoré sa toto minimum nadobúda, označme

$$A_\alpha(\mathbf{x}) = \arg \min_{\zeta \in \mathbb{R}} F_\alpha(\mathbf{x}, \zeta),$$

je neprázdny, uzavretý, ohraničený interval a  $\alpha$ -VaR straty je

$$\zeta_\alpha(\mathbf{x}) = \{\min \gamma : \gamma \in A_\alpha(\mathbf{x})\} \text{ a platí } \phi_\alpha(\mathbf{x}) = F_\alpha(\mathbf{x}, \zeta_\alpha(\mathbf{x})).$$

**Tvrdenie 4** Minimalizovanie miery  $\alpha$ -CVaR vzhľadom k  $\mathbf{x} \in X$  je ekvivalentné minimalizovaniu  $F_\alpha(\mathbf{x}, \zeta)$  vzhľadom k  $(\mathbf{x}, \zeta) \in X \times \mathbb{R}$  v tom zmysle, že

$$\min_{\mathbf{x} \in X} \phi_\alpha(\mathbf{x}) = \min_{(\mathbf{x}, \zeta) \in X \times \mathbb{R}} F_\alpha(\mathbf{x}, \zeta),$$

a navyše, v bode  $(\mathbf{x}^*, \zeta^*)$  nadobúda pravá strana minimum práve vtedy, keď v bode  $\mathbf{x}^*$  nadobúda minima ľavá strana a  $\zeta^* \in A_\alpha(\mathbf{x}^*)$ . Ak je množina  $X \subset \mathbb{R}^n$  konvexná a  $f(\mathbf{x}, \mathbf{y})$  konvexná vzhľadom k  $\mathbf{x}$ , potom je  $F_\alpha(\mathbf{x}, \zeta)$  konvexná vzhľadom k  $(\mathbf{x}, \zeta)$  a  $\phi_\alpha(\mathbf{x})$  je konvexná vzhľadom k  $\mathbf{x}$ .

Vzhľadom k tomuto tvrdeniu, není potrebné k určaniu rozhodovacieho vektora, pre ktorý  $\alpha$ -CVaR nadobúda minimálnu hodnotu, pracovať priamo s funkciou  $\phi_\alpha(\mathbf{x})$ , ktorá takisto zahrňuje výpočet hodnoty  $\alpha$ -VaR. Namiesto toho môžeme priamo pracovať s funkciou  $F_\alpha(\mathbf{x}, \zeta)$ , ktorá má atraktívnejšie vlastnosti. Dôkazy týchto tvrdení čitateľ nájde v [12].

Ďalej budeme predpokladať, že strata  $f(\mathbf{x}, \mathbf{y})$  a očakávaný výnos  $R(\mathbf{x})$  sú lineárne funkcie premennej  $\mathbf{x}$ , preto podľa tvrdenia (4) je miera rizika CVaR  $\phi_\alpha(\mathbf{x})$  konvexná vzhľadom k  $\mathbf{x}$ . Máme teda splnené predpoklady tvrdenia (2). Znamená to, že maximalizovaním výnosu pri ohraničení rizika CVaR generujeme rovankú efektívnu množinu, ako pri minimalizovaní rizika CVaR pri garantovaní minimálneho výnosu. Následovne uvedieme dve dôležité tvrdenia potrebné ku generovaniu efektívnej množiny. V texte ďalej označme  $R(\mathbf{x})$  očakávaný výnos portfólia  $\mathbf{x}$ .

**Tvrdenie 5** Optimalizačné problémy

$$\begin{aligned} (P1) \quad & \min_{\mathbf{x} \in X} -R(\mathbf{x}), \quad \phi_\alpha(\mathbf{x}) \leq \omega, \\ (P2) \quad & \min_{(\mathbf{x}, \zeta) \in X \times \mathbb{R}} -R(\mathbf{x}), \quad F_\alpha(\mathbf{x}, \zeta) \leq \omega, \end{aligned}$$

majú rovnakú optimálnu hodnotu. Ak ohraničenie v úlohe P1 je aktívne, tak  $(\mathbf{x}^*, \zeta^*)$  minimalizuje úlohu P2 práve vtedy, keď  $\mathbf{x}^*$  minimalizuje úlohu P1 a  $\zeta^* \in A_\alpha(\mathbf{x}^*)$ . Špeciálne, ak interval  $A_\alpha(\mathbf{x}^*)$  je jednobodvá množina, tak  $\zeta^*$  je  $\alpha$ -VaR portfólia  $\mathbf{x}^*$ .

**Tvrdenie 6** Optimalizačné problémy

$$\begin{aligned} (P3) \quad & \min_{\mathbf{x} \in X} \phi_\alpha(\mathbf{x}) - \mu R(\mathbf{x}), \quad \mu \geq 0 \\ (P4) \quad & \min_{(\mathbf{x}, \zeta) \in X \times \mathbb{R}} F_\alpha(\mathbf{x}, \zeta) - \mu R(\mathbf{x}), \quad \mu \geq 0 \end{aligned}$$

majú rovnakú optimálnu hodnotu. Úloha P4 nadobúda v bode  $(\mathbf{x}^*, \zeta^*)$  optimum práve vtedy, ak úloha P3 nadobúda v bode  $\mathbf{x}^*$  optimum a  $\zeta^* \in A_\alpha(\mathbf{x}^*)$ . Špeciálne, ak interval  $A_\alpha(\mathbf{x}^*)$  je jednobodová množina, tak  $\zeta^*$  je  $\alpha$ -VaR portfólia  $\mathbf{x}^*$ .

Dôkazy napríklad v Krokmal a kol. [6].

Vyššie uvedené tvrdenia a vhodný spôsob aproximácie funkcie  $F_\alpha(\mathbf{x}, \zeta)$  nám umožňujú skonštruovať optimalizačný problém. Ak napríklad, vektory  $\mathbf{y}_1, \dots, \mathbf{y}_J$  sú generované čísla, alebo historické dáta náhodnej veličiny s hustotou  $p(\mathbf{y})$ , tak jedna z možných aproximácií funkcie

$$F_\alpha(\mathbf{x}, \zeta) = \zeta + (1 - \alpha)^{-1} \int_{\mathbf{y} \in \mathbb{R}^n} [f(\mathbf{x}, \mathbf{y}) - \zeta]^+ p(\mathbf{y}) d\mathbf{y}$$

je

$$\hat{F}_\alpha(\mathbf{x}, \zeta) = \zeta + (1 - \alpha)^{-1} \sum_{j=1}^J \pi_j [f(\mathbf{x}, \mathbf{y}_j) - \zeta]^+,$$

kde  $\pi_j$  je pravdepodobnosť scénara  $\mathbf{y}_j$ ,  $j = 1, \dots, J$ . Ak je funkcia straty  $f(\mathbf{x}, \mathbf{y})$  lineárna v  $\mathbf{x}$ , potom funkcia  $\hat{F}_\alpha(\mathbf{x}, \zeta)$  je konvexná, a po častiach lineárna.

Ďalej, ak použijeme pomocné (doplňkové) premenné, funkciu  $\hat{F}_\alpha(\mathbf{x}, \zeta)$  môžeme nahradiť lineárnou funkciou a sadou lineárnych ohraničení:

$$\begin{aligned} & \zeta + (1 - \alpha)^{-1} \sum_{j=1}^J \pi_j z_j \\ z_j & \geq f(\mathbf{x}, \mathbf{y}_j) - \zeta, \quad z_j \geq 0, \quad j = 1, \dots, J, \quad \zeta \in \mathbb{R}. \end{aligned}$$

Napríklad ohraničenie  $\phi_\alpha(\mathbf{x}) \leq \omega$ , ktoré je v optimalizačnej úlohe ekvivalentné  $F_\alpha(\mathbf{x}, \zeta) \leq \omega$ , môžeme nahradiť sadou ohraničení

$$\begin{aligned} & \zeta + (1 - \alpha)^{-1} \sum_{j=1}^J \pi_j z_j \leq \omega, \\ z_j & \geq f(\mathbf{x}, \mathbf{y}_j) - \zeta, \quad z_j \geq 0, \quad j = 1, \dots, J, \quad \zeta \in \mathbb{R}. \end{aligned}$$

Podobný princíp môžeme použiť pri ekvivalentných formuláciách úlohy optimálneho portfólia.

## Základný model s ohraničeniami

Zhrňme vyššie uvedené skutočnosti do optimalizačného modelu maximalizovania očakávaného výnosu s obmedzením rizika CVaR. Portfólio prezentujeme počtom držaných jednotlivých aktív. Investor rebalancuje svoje súčasné portfólio  $\mathbf{x}^0$  so súčasnými cenami jeho aktív  $\mathbf{q} = (q_1, \dots, q_n)^\top$  a tržnou hodnotou teda  $\mathbf{q}^\top \mathbf{x}^0$ . Nech náhodný vektor  $\mathbf{y} = (y_1, \dots, y_n)^\top$  označuje ceny aktív na konci periódy. Stratu na konci periódy reprezentujeme funkciou

$$f(\mathbf{x}, \mathbf{y}, \mathbf{x}^0, \mathbf{q}) = -\mathbf{y}^\top \mathbf{x} + \mathbf{q}^\top \mathbf{x}^0,$$

takže očakávaná hodnota portfólia na konci periódy je  $R(\mathbf{x}) = -\mathbb{E}f(\mathbf{x}, \mathbf{y}) + \mathbf{q}^\top \mathbf{x}^0$ . Obmedzenie na riziko budeme reprezentovať koeficientom  $\omega$ , predstavujúci časť hodnoty portfólia vystavenú riziku:

$$\phi_\alpha(\mathbf{x}) \leq \omega \mathbf{q}^\top \mathbf{x}^0.$$

Sada linearizovaných ohraničení je

$$\begin{aligned} & \zeta + (1 - \alpha)^{-1} \sum_{j=1}^J \pi_j z_j \leq \omega \sum_{i=1}^n q_i x_i^0, \\ z_j & \geq \sum_{i=1}^n (-y_{ij} x_i + q_i x_i^0) - \zeta, \quad z_j \geq 0, \quad j = 1, \dots, J. \end{aligned}$$

Konečne, finálny lineárny program s dodatočnými ohraničeniami je:

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{u}^+, \mathbf{u}^-, \zeta} \sum_{i=1}^n -\mathbb{E}y_i x_i \\
& \zeta + (1 - \alpha)^{-1} \sum_{j=1}^J \pi_j z_j \leq \omega \sum_{k=1}^n q_k x_k^0 \\
& z_j \geq i = 1 \sum^n (-y_{ij} x_i + q_i x_i^0) - \zeta, & z_j \geq 0, \quad j = 1, \dots, J \\
& q_i x_i \leq \delta_i \sum_{k=1}^n q_k x_k, & i = 1, \dots, n \\
& q_i x_i \geq \varepsilon_i \sum_{k=1}^n q_k x_k, & i = 1, \dots, n \\
& \sum_{j \in G_g} x_j q_j \leq \vartheta_g \sum_{j=1}^n x_j q_j, & g = 1, \dots, G \\
& \sum_{j \in G_g} x_j q_j \geq \xi_g \sum_{j=1}^n x_j q_j, & g = 1, \dots, G \\
& \sum_{i=1}^n (u_i^+ + u_i^-) q_i \leq \Theta \\
& \sum_{i=1}^n q_i x_i^0 = \sum_{i=1}^n c_i q_i (u_i^+ + u_i^-) + \sum_{i=1}^n q_i x_i \\
& x_i - x_i^0 = u_i^+ - u_i^-, & i = 1, \dots, n \\
& 0 \leq u_i^- \leq u_i^s, \quad 0 \leq u_i^+ \leq u_i^b, & i = 1, \dots, n.
\end{aligned} \tag{4.10}$$

Nie je celkom pravda, že tento problém je rýdzo lineárny. Premenná  $\zeta$  je totiž neohraničená. Rozložením tejto premnnej na kladnú a zápornú časť  $\zeta = \zeta^+ - \zeta^-$ , kde  $\zeta^-, \zeta^+ \geq 0$  a pridaním do modelu však dostaneme lineárny program.

## 4.2 Numerické ilustrácie

V tejto časti budeme ilustrovať vyššie popísané optimalizačné úlohy. K ilustráciám použijeme software *Wolfram Mathematica*<sup>®</sup>, verzie 6.0.1.0 a počítač s procesorom Intel Core 2 Duo 2.00 GHz a 2GB RAM pamäte. Pokiaľ neuvedieme iný zdroj, všetky dáta budeme čerpať zo serveru *Yahoo Finance* (<http://finance.yahoo.com/>), prostredníctvom zabudovanej funkcie `FinancialData`. Naprogramované procedúry a súbory, na ktoré sa odkazujeme, sú uložené na priloženom CD nosiči.

### 4.2.1 Numerické optimalizačné metódy

V optimalizačných procedúrach ako optimalizátor použijeme zabudované funkcie `NMinimize`, `LinearProgramming` a `FindMinimum`.

Prvá funkcia `NMinimize`, numericky minimalizuje reálnu funkciu so snahou nájsť globálne minimum. Táto funkcia pracuje práve s minimalizačnými metódami popísanými v kapitole 3. Z tohto dôvodu tu nemôžeme hovoriť o získaní presného globálneho minima. Dokáže dobre pracovať aj s nelineárnymi ohraničeniami využitím penalizačnej funkcie, ktorú volí automaticky. Pri lineárnej účelovej funkcii a lineárnych ohraničeniach globálne minimum nájde vždy. Okrem metód Nelder-Mead, simulovaného tavenia a diferenciálnych evolúcií, pracuje aj s metódou náhodného hľadania minima. Výhodou je, že software umožňuje zvoliť metódu automaticky podľa charakteru úlohy. Parametre v týchto metódach sú vybrané buď automaticky, alebo ich môžeme príkazom meniť.

Funkcia `LinearProgramming` rieši úlohu lineárneho programovania simplexovou metódou, alebo túto úlohu numericky minimalizuje užitím metódy vnútorného bodu. Vstupný lineárny program nemusí byť navyše v štandardnej forme (1.3), čo značne uľahčuje prácu pri programovaní procedúr. Optimalizačnú metódu možno vybrať ľubovoľne, alebo ponechať automatický výber.

Funkcia FindMinimum numericky hľadá lokálne minimum ohraničeného problému. Pri použití diferencovateľných funkcií v ohraničeniach a diferencovateľnej účelovej funkcie, môžeme ako optimalizačnú numerickú metódu v nastavení zvoliť metódu vnútorného bodu. Túto funkciu použijeme na riešenie úlohy kvadratického programovania. Ako uvidíme neskôr, účelová funkcia tohto problému bude konvexná, takže pomocou tejto funkcie budeme hľadať globálne minimum.

## 4.2.2 Optimalizačné procedúry

V ilustráciach uvažujme zatiaľ portfólio zložené z vybraných akcií obsiahnutých v indexe *Standard and Poor's 100 Stock Index*. Ako zdroj historických dát použijeme obchodovacie ceny týchto aktív v období rokov 1990-2007. Períodu volíme jeden mesiac, takže v tomto prípade použijeme ceny v prvom obchodovacom dni daných mesiacov. Do výberu akcií však zahrnieme len tie, ktoré boli obchodované (majú záznam v zdroji) vo všetkých týchto obchodovacích dňoch, aby uvažované dáta boli konzistentné. V priloženom programe však možno obdobie zdroja dát a periódu meniť.

Bez ujmy na všeobecnosti, označme prvý obchodovací deň mesiaca január roku 1990 časom  $t = 0$  a prvý obchodovací deň mesiaca december roku 2007 časom  $t = T$ . Označme  $n$  uvažovaných aktív symbolom  $i$ ,  $i = 1, \dots, n$  a cenu  $i$ -tého aktíva v čase  $t$ ,  $t = 0, \dots, T$  symbolom  $p_{it}$ ,  $i = 1, \dots, n$ . Miera výnosnosti  $r_{it}$  aktíva  $i$  v perióde  $t$  je

$$r_{it} = \frac{p_{it} - p_{it-1}}{p_{it-1}}, \quad t = 1, \dots, T, \quad i = 1, \dots, n.$$

Očakávané miery výnosov, ich rozptyl a kovarianciu nahradíme výberovými odhadmi. Historické dáta s väčšou uvažovanou periódou majú často slabú predpovedaciu hodnotu. Avšak, na ilustráciu daných metód nám to postačuje. Navyše tieto modely možno jednoducho kombinovať s vhodnejšími štatistickými odhadmi, metódami Monte Carlo a podobne.

Ďalej sa budeme venovať hľadaniu optimálneho portfólia pri rôznych mierach rizika a tak isto do úvahy zoberieme aj ďalšie ohraničenia. Optimálny výber pri použití modeli, ak to dovoľí rozsah, ilustrujeme pomocou grafu reprezentujúceho efektívnu množinu. Je dôležité zdôrazniť, že uvedené príklady slúžia ako ilustrácie použitých metód a prístupov, a nemožno ich použiť ako doporučenia v reálnych obchodných situáciach. Tie si vyžadujú ďaleko sofistikovanejšie prístupy, na ktoré v tejto práci nie je priestor. Pre nasledujúce ilustrácie optimalizačných procedúr sme zvolili 7 náhodne vybraných akcií. Zvolený počet je relatívne malý, ale pre znázornenie postačujúci. Generovanie efektívnej množiny si totiž vyžaduje zrábanie značného počtu optím a pri menšom počte inštrumentov je to ešte časovo únosné. O portfóliach s väčším počtom inštrumentov pojednáme neskôr.

Tabuľka 4.1 obsahuje naše vybrané akcie a ich základné charakteristiky: názov, sektor podnikania, očakávanú mieru výnosu, smerodajnú odchýlku a obchodovaciu cenu v čase  $T$ . Odhady očakávaných výnosov a smerodajných odchýliek sú výberové. Spolu s týmito rizikovými aktívami do portfólia zahrnieme ešte bezrizikové aktívum  $x_f$  (hotovosť) a budeme predpokladať, že má konštantnú mesačnú mieru výnosu 0.0025,  $\rho_f = 0.0025$ . Pre kritérium optimality sme volili očakávaný výnos a vybrané miery rizika. Prístup k meraniu rizika CVaR a VaR volíme ako *neparametrický* (pozri aj 4.1.3), tak aj *parametrický*.

V parametrickom prístupe vychádzame z predpokladu, že miery výnosov  $\rho_i$  aktív  $i = 1, \dots, n$  majú normálne rozdelenia  $N(\mu_i, \sigma_i^2)$  s kovariančnou maticou  $\mathbf{Q}$ ,  $\mathbf{Q}_{ij} = \sigma_{ij}$ . Potom

STOCK	SECTOR	EXP RET %	ST DEV %	PRICE (\$)
Emc Corporation	Computers	3.70473	15.7326	13.2
Citigroup Inc	Banks	2.19665	8.88184	50.48
Williams Companies Inc	GasProduction	2.05498	12.62	25.55
Bank Of New York Company Inc	Banks	1.99735	9.47198	39.83
Home Depot Inc	HomeImprovement	1.9573	8.4951	38.13
Baker Hughes Inc	ConstructionMining	1.2877	9.95215	73.44
The Coca Cola Company	Beverages	1.20354	6.57021	46.11

Tabulka 4.1: Zoznam zahrnutých akcií do portfólia so základnými charakteristikami.

mera výnosu  $\rho_{\mathbf{x}}$  pre portfólio  $\mathbf{x}$  (váhový vektor) má približne normálne rozdelenie

$$\rho_{\mathbf{x}} \sim N(\boldsymbol{\mu}^{\top} \mathbf{x}, \mathbf{x}^{\top} \mathbf{Q} \mathbf{x}), \quad \boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^{\top},$$

takže VaR a CVaR možno ľahko určiť vďaka tomuto rozdeleniu. Odhady pre parametre  $\boldsymbol{\mu}$  a  $\mathbf{Q}$  sme zvolili výberové, na základe historických mier výnosov  $r_{it}$ ,  $i = 1, \dots, n$ ,  $t = 1, \dots, T$ . Keďže poznáme rozdelenia (hustoty) výnosov, softvér Mathematica nám umožňuje zrátať rizika VaR a CVaR pre portfólio  $\mathbf{x}$  podľa tohto rozdelenia (pozri napríklad Hurt [5]).

V neparametrickom prístupe odhadujeme príslušné miery rizík priamo pomocou historických dát alebo scénarov. Nech  $L_s(\mathbf{x})$  označuje stratu portfólia  $\mathbf{x}$  v scenári (historické ceny môžeme chápať ako scénare)  $s$ ,  $s = 1, \dots, S$ . Napríklad v našom prípade je strata portfólia  $\mathbf{x}$  v perióde (scenári)  $s$ ,  $s = 1, \dots, T$

$$L_s(\mathbf{x}) = -\mathbf{r}_s^{\top} \mathbf{x}, \text{ kde } \mathbf{r}_s = (r_{1s}, \dots, r_{ns})^{\top}.$$

Usporiadajme tieto straty podľa ich hodnôt tak, že  $L_1 \leq L_2 \leq \dots \leq L_S$ . Takže pre odhad rizika VaR a CVaR portfólia  $\mathbf{x}$  na hladine  $\alpha$ , označme  $\text{VaR}_{\alpha}(\mathbf{x})$  a  $\text{CVaR}_{\alpha}(\mathbf{x})$  máme vzťahy:

$$\text{VaR}_{\alpha}(\mathbf{x}) = L_{[\alpha S]}, \quad (4.11)$$

$$\text{CVaR}_{\alpha}(\mathbf{x}) = \frac{\sum_{i=1}^S L_i \mathbf{1}_{(L_i > \text{VaR}_{\alpha}(\mathbf{x}))}}{\sum_{i=1}^S \mathbf{1}_{(L_i > \text{VaR}_{\alpha}(\mathbf{x}))}}, \quad (4.12)$$

kde  $[x]$  je celá časť čísla  $x \in \mathbb{R}$  a  $\mathbf{1}_{(v)} = 1$  ak je  $v$  pravda a 0 ak je  $v$  nepravda.

Teraz podrobnejšie preskúmame optimálne portfólia pri rôznych mierach rizika. Všetky prezentované výsledky a naprogramované výpočetné procedúry čitateľ nájde v priloženom súbore. Vstupné parametre procedúr sú historické ceny akcií a ďalšie parametre potrebné k charakterizácii rizika a dodatočných ohraňení. Keďže uvažujeme aj transakčné náklady a ďalšie ohraňenia hodnôt pri rebalancovaní portfólia, vstupom je aj portfólio držané v súčasnosti  $\mathbf{x}^{in}$  ktoré potrebujeme rebalancovať. Zdôraznime, že portfólio je v procedúrach prezentované počtom držaných aktív, a nie podielmi k celkovej hodnote portfólia.

Pre vstupné parametre dodatočných ohraňení použijeme označenia zavedené v (4.1.1). V príkladoch uvažujeme pre rizikové aktíva  $i = 1, \dots, n$  a bezrizikové aktívum  $i = n + 1$  (hotovosť) lineárne transakčné náklady ( $\mathbf{c}$ ), minimálne a maximálne povolené časti kapitálu

investované do jednotlivých aktív ( $\boldsymbol{\varepsilon}, \boldsymbol{\delta}$ ), ohraničenia likvidity ( $\mathbf{u}^s, \mathbf{u}^b$ ) a maximálny povolený obrat rebalancovania portfólia ( $\Theta$ ). Parametre ohraňení pre jednotlivé aktíva vstupujú teda do procedúr vo vektorovom tvare:

$$\begin{aligned}
\mathbf{c} &= (c_1, \dots, c_{n+1})^\top \\
\boldsymbol{\varepsilon} &= (\varepsilon_1, \dots, \varepsilon_{n+1})^\top \\
\boldsymbol{\delta} &= (\delta_1, \dots, \delta_{n+1})^\top \\
\mathbf{u}^s &= (u_1^s, \dots, u_{n+1}^s)^\top \\
\mathbf{u}^b &= (u_1^b, \dots, u_{n+1}^b)^\top.
\end{aligned} \tag{4.13}$$

Všetky naprogramované procedúry v priloženom súbore umožňujú implementovať tieto ohraňenia do riešených problémov.

V niektorých príkladoch namiesto rebalancovania portfólia popisujeme investovanie počiatočného kapitálu (hotovosti) do akcií, alebo neuvažujeme dodatočné ohraňenia. Toto však možno ľahko dosiahnuť vhodnou voľbou vstupných parametrov a počiatočného portfólia. V nasledujúcich ukázkach tak napríklad v súčasnom portfóliu máme \$10 000 a žiadne rizikové aktívum. Všetky tieto prostriedky chceme vložiť do rizikových aktív, pričom nemáme žiadne ďalšie obmedzenia a neuvažujeme transakčné náklady. Vstupné parametre týchto ohraňení budú preto v nasledujúcom tvare:

$$\begin{aligned}
c_i &= 0, & i &= 1, \dots, n+1 \\
\varepsilon_i &= 0, & i &= 1, \dots, n+1 \\
\delta_i &= 1, & i &= 1, \dots, n, \\
\delta_{n+1} &= 0 \\
u_i^s &= \infty, & i &= 1, \dots, n \\
u_i^b &= \infty, & i &= 1, \dots, n \\
x_i^{in} &= 0, & i &= 1, \dots, n \\
x_{n+1}^{in} &= 10\,000, \\
\Theta &= \infty.
\end{aligned} \tag{4.14}$$

Symbol nekonečna  $\infty$  v ohraňeniach pri spúšťaní procedúr nahradíme vhodne zvoleným číslom.

Ďalej predstavíme základné výsledky dosiahnuté pri optimalizácii portfólia pri použití rôznych mier rizík a optimalizačných metód. Uvažujeme teda 7 rizikových aktív a hotovosť bez dodatočných ohraňení.

### Markowitzov Model (Mean-Variance)

Procedúra hľadá portfólio s minimálnou smerodajnou odchýlkou a požadovaným očakávaným výnosom aspoň  $\mu_p$  (vstupný parameter). Tento prístup je samozrejme parametrický. Za predpokladu normálneho rozdelenia miery výnosu, sú úlohy minimalizovania rizika ako smerodajnej odchýlky, VaR a CVaR pri požadovanom očakávanom výnose ekvivalentné. To znamená, že pri požiadavke minimálneho výnosu aspoň  $\mu_p$  je optimálne portfólio pri každej takto uvažovanej miere rizika rovnaké (pozri napríklad [12]). Výstupom procedúry je dosiahnuté optimálne portfólio, očakávaný výnos, smerodajná odchýlka a (ne)parametrické miery rizík VaR a CVaR na hladine  $\alpha$  (vstupný parameter). Táto procedúra pomocou

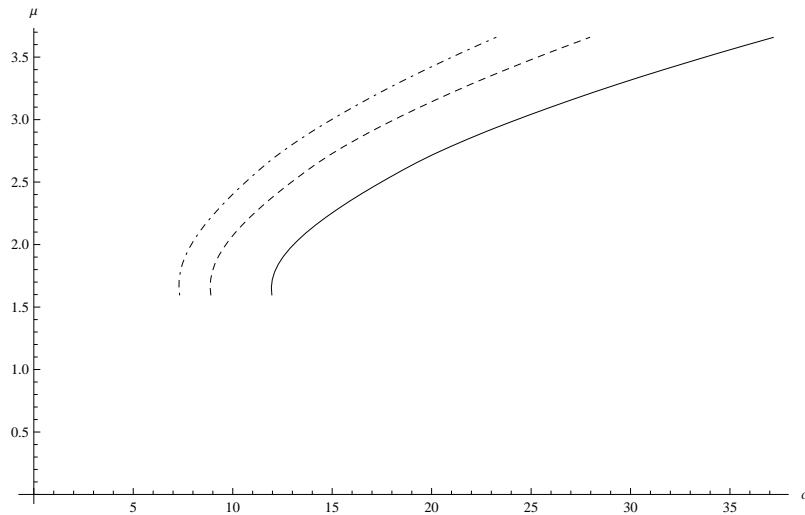


funkcie FindMinimum hľadá globálne minimum metódou vnútorného bodu pre úlohu kvadratického programovania (2.3). Keďže kovariančná matica výnosov je pozitívne semidefinitná, smerodajná odchýlka je konvexná funkcia. Treba ešte pripomenúť, že ohraničenia pre transakčné náklady (4.2) nie sú lineárne (obsahujú absolútnu hodnotu). Tomu sa však vyhneme zavedením doplnkových premenných  $\mathbf{p}^+ = (p_1^+, \dots, p_n^+)^T$ ,  $\mathbf{p}^- = (p_1^-, \dots, p_n^-)^T$  a prevedením týchto ohraničení na nasledujúci lineárny systém:

$$\begin{aligned} \sum_{i=1}^n q_i x_i^0 &= \sum_{i=1}^n c_i q_i (p_i^+ + p_i^-) + \sum_{i=1}^n q_i x_i \\ \mathbf{x} - \mathbf{x}^0 &= \mathbf{p}^+ - \mathbf{p}^- \\ \mathbf{p}^+ &\geq \mathbf{0}, \quad \mathbf{p}^- \geq \mathbf{0}. \end{aligned}$$

Týmto nám ale do problému pribudne  $2n$  nových premenných.

Uvažujme investovanie určitého kapitálu (hotovosti) do uvažovaných akcií podľa základného Markowitzovho modelu (4.1) bez dodatočných ohraničení. Efektívnu množinu skonštruujeme aplikovaním procedúry pre 20 rôznych hodnôt minimálneho požadovaného výnosu portfólia  $\mu_p$ . Tieto hodnoty volíme v rozmedzí minimálneho a maximálneho očakávaného výnosu z množiny uvažovaných akcií. Obrázok 4.1 zobrazuje tieto dosiahnuté výsledky v implementácii rizika CVaR pre tri rôzne hladiny parametra  $\alpha$ .



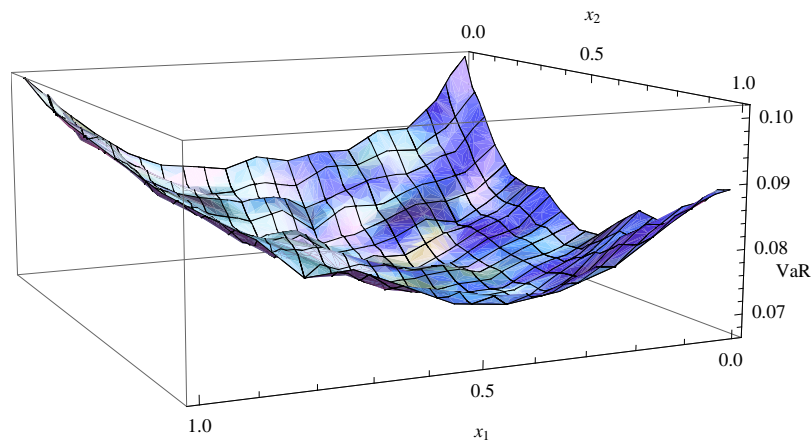
Obrázok 4.1: Dosiahnuté výsledky Mean-Variance modelu v implementácii rizika CVaR pre hladiny  $\alpha = 0.99$  (plne),  $\alpha = 0.95$  (čiarkovane) a  $\alpha = 0.9$  (bodkočiarkovane).

## VaR

Keďže za predpokladu normálnych rozdelení výnosov, sú úlohy minimalizovania smerodajnej odchýlky a rizika VaR ekvivalentné, pozornosť venujeme neparametrickému prístupu na základe historických dát alebo scénarov. Procedúra v priloženom súbore minimalizuje pomocou funkcie NMinimize neparametrickú mieru rizika VaR (4.11) na hladine  $\alpha$  (vstupný parameter) s požadovaným očakávaným výnosom portfólia  $\mu_p$  (vstupný parameter). Zdôraznime, že na rozdiel od ostatných procedúr tu požadujeme presný očakávaný

výnos. Výstupom procedúry je dosiahnuté optimálne portfólio, očakávaný výnos, smerodajná odchýlka a (ne)parametrické miery rizík VaR a CVaR na hladine  $\alpha$ .

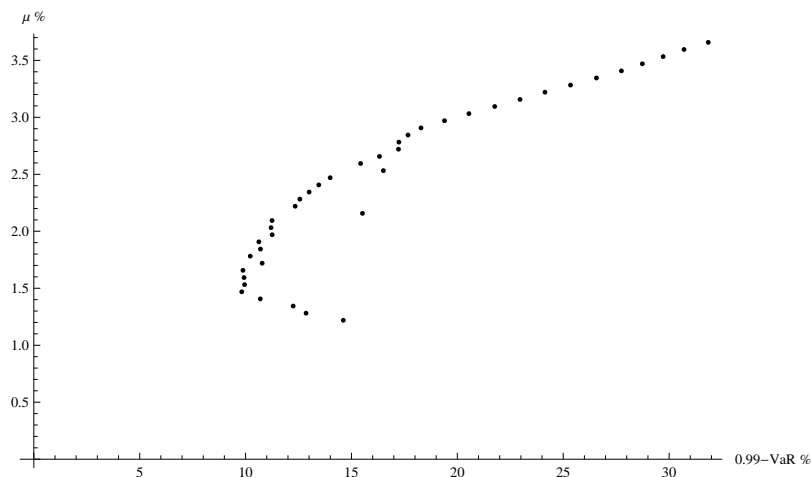
Ako sme už hovorili, neparametrická miera rizika VaR počítaná na základe scénarov alebo historických dát je nekonvexnou funkciou so značným počtom lokálnych extrémov. Túto skutočnosť ilustrujeme na grafe tejto funkcie pre tri náhodne zvolené akcie a generovaním 500 scénarov multinormálneho rozdelenia s parametrami im odpovedajúcich očakávaných mier výnosov a kovariančnej matice (obrázok 4.2). Pri portfóliach väčších rozmerov si môžeme predstaviť, ako je numericky náročné nájsť globálne minimum takejto funkcie. K minimalizovaniu tejto miery rizika je preto vhodné použiť jednu z metód simulovaného tavenia alebo diferenciálnych evolúcií.



Obrázok 4.2: Nekonvexnosť neparametrickej miery VaR. Osy  $x_1$  a  $x_2$  prezentujú podiel investovaný do prvých dvoch akcií ( $x_3 = 1 - x_1 - x_2$ ) a osa VaR odpovedajúcu mieru na hladine  $\alpha = 0.99$ .

Túto procedúru spustíme pre 20 rôznych hodnôt požadovaného výnosu  $\mu_p$  na hladine  $\alpha = 0.99$  s automatickou voľbou optimalizačnej metódy. Uvažujeme len investovanie do rizikových aktív bez dodatočných ohraničení. Výsledky tohto procesu sú prezentované na obrázku 4.3. Všimli sme si, že niektoré dosiahnuté výsledky nereprezentujú efektívne portfólio. Dostali sme totiž body, ktoré pri menšom očakávanom výnose majú väčšie riziko VaR ako iný bod s väčším očakávaným výnosom. Numerické optimalizačné algoritmy totiž nedosiahli globálneho minima. Chceli sme tým upozorniť, že skutočne nie je ľahké získať globálne minimum funkcie s takýmto tvarom. O zlepšení tohto optimalizačného procesu budeme ešte neskôr hovoriť.

Treba ešte upozorniť na jednu dôležitú skutočnosť. Portfólia pri nízkom požadovanom výnose sú síce neefektívne, ale ako uvidíme neskôr, je to spôsobené neefektivitou nášho trhu. Niektoré akcie s nižším očakávaným výnosom majú väčšie riziko. Nemusí sa teda nutne jednať o nesprávne dosiahnuté globálne minimum. Túto skutočnosť sme nepostrehli pri Markowitzovom modeli preto, že sme v ňom požadovali výnos väčší alebo rovný požadovanému výnosu. To, prečo v procedúre pre riziko VaR požadujeme presný očakávaný výnos, zdôvodníme neskôr.



Obrázek 4.3: Dosiahnuté výsledky neparametrickeho Mean-VaR modelu na hladine  $\alpha = 0.99$ .

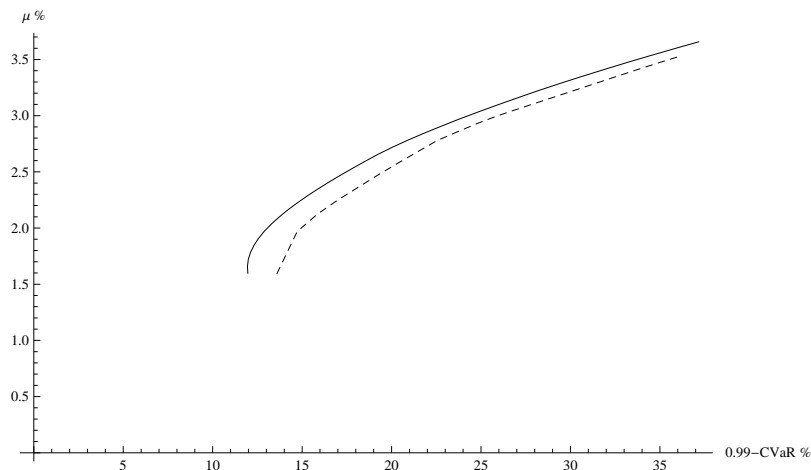
## CVaR

Priame minimalizovanie neparametrickej miery rizika CVaR (4.12) sa testovaním ukázalo ako nevýhodné. Problém spočíva v tom, že optimalizačná procedúra potrebuje pri rátaní funkčnej hodnoty v každom bode (portfóliu) najprv spočítať riziko VaR (usporiadať hodnoty strát pre jednotlivé scénare). Spolu s ďalšími ohraničeniami a komplikovanou voľbou penalizačnej funkcie je tento proces neefektívny. Dobrou voľbou sa ukazuje použitie teoretický aparát prebraný v 4.1.3 a zabudovanú funkciu LinearProgramming. V tomto prípade maximalizujeme očakávaný výnos portfólia a požadujeme, aby riziko CVaR na hladine  $\alpha$  nepresiahlo časť  $\omega$  hodnoty portfólia. Priložená procedúra pracuje podľa lineárneho modelu 4.10 so vstupnými parametrami  $\alpha$  a  $\omega$  a predpokladá rovnakú pravdepodobnosť pre všetky scénare, to je  $\pi_i = 1/J$ ,  $i = 1, \dots, J$ . K ilustrácii efektívnej množiny spustíme optimalizačnú procedúru pre 20 vhodne zvolených hodnôt parametra  $\omega$ . Obrázok 4.4 znázorňuje tieto dosiahnuté výsledky použitím tohto lineárneho modelu v porovnaní s dosiahnutými výsledkami procedúry podľa Markowitzovho modelu, v implementácii rizika CVaR na hladine  $\alpha = 0.99$ .

Na dolnej ose sú vlastne volené hranice  $\omega$  tolerancie rizika CVaR. Pri voľbe  $\omega = 0.2$  tak napríklad maximalizujeme očakávaný výnos a požadujeme, aby riziku CVaR bolo vystavené maximálne 20% hodnoty portfólia. Výhodou tohto prístupu je optimalizácia prostredníctvom lineárneho programovania. Použitie metódy vnútorného bodu nám umožňuje zahrnúť do modelu vysoký počet aktív a historických dát alebo scénarov.

## Kono-Yamazaki metóda

Optimalizujeme ešte naše portfólio pri riziku strednej absolútnej odchýlky. Optimalizačná procedúra pracuje podľa lineárneho modelu (4.1.2), teda minimalizuje strednú absolútnu odchýlku s požadovaným očakávaným výnosom aspoň  $\mu_p$ . Pri generovaní efektívnej množiny sme dostali takmer zhodnú krivku, ako pri minimalizovaní smerodajnej odchýlky (v Markowitzovom modeli). To potvrdzuje fakt, že výnosy akcií majú približne normálne roz-



Obrázek 4.4: Dosiahnuté výsledky Mean-Variance modelu (plne) a lineárneho Mean-CVaR modelu (čiarkovane) v implementácii rizika CVaR na hladine  $\alpha = 0.99$ .

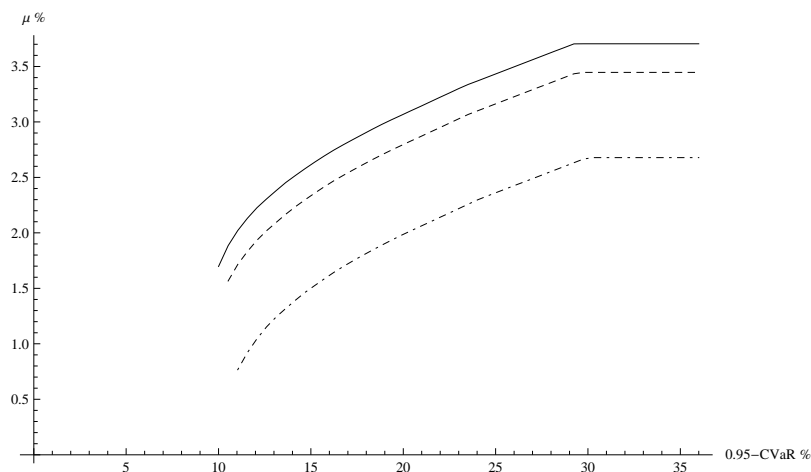
delenia. Navyše, tieto výsledky sme dostali bez odhadu kovariančnej matice. Túto procedúru možno spustiť s použitím generovaných scénarov, ktoré umožňujú zahrnúť do modelu extrémne hodnoty strát (stress testing). Graf reprezentujúci efektívnu množinu tohto a Markowitzovho modelu v implementácii rizika smerodajnej odchýlky je v priloženom súbore, kde nám po patričnom zväčšení poskytuje zreteľnejšie porovnanie týchto modelov.

### 4.2.3 Ohraničenia v modeloch

V ilustráciach vyššie sme neuvažovali dodatočné ohraničenia. Portfólio sme jednak obmedzili na investované peniaze, zakázali sme krátke pozície a požadovali sme istý očakávaný výnos. Do optimalizačného procesu skúsime pridať aj ďalšie dodatočné ohraničenia. Uvažujme teda minimalizovanie rizika CVaR na hladine  $\alpha = 0.95$  pomocou lineárneho modelu. Naše súčasné portfólio obsahuje len hotovosť a chceme ju celú investovať do rizikových aktív, ktoré vyžadujú transakčné náklady. Transakčné náklady na hotovosť neuvažujeme. Obrázok 4.5 opisuje túto situáciu pri transakčných nákladoch  $c = 0\%$ ,  $c = 0.25\%$  a  $c = 1\%$ . Môžeme si povšimnúť, že tieto lineárne náklady znižujú očakávaný výnos nelineárne a majú podstatný vplyv na voľbu portfólia.

Venujme sa teraz ohraničeniam hodnôt. Konkrétne uvažujme o investovaní počiatočného kapitálu (hotovosti) do zmiených 7 rizikových aktív (akcií) a bezrizikového aktíva s konštantným mesačným výnosom  $0.25\%$ . Investor je pritom obmedzený na investovanie do jednej akcie maximálne  $40\%$  celkovej hodnoty portfólia a maximálne  $20\%$  do bezrizikového aktíva. Optimalizujme portfólio tohto investora. Budeme maximalizovať očakávaný výnos pričom riziku  $0.99$ -CVaR vystavíme maximálne časť  $\omega$  celkovej hodnoty portfólia. Použijeme teda skôr diskutovaný lineárny model. Sledujme teraz diverzifikáciu tohto portfólia pri rôznych hodnotách parametra  $\omega$ . Tabuľka 4.2 znázorňuje podiely (váhy) investované do jednotlivých rizikových aktív a hotovosti pri rôznych vystaveniach riziku.

Pri zmenšovaní parametra  $\omega$  vystavujeme toto portfólio menšiemu riziku. Je pekne vidieť ako postupne presúvame kapitál z rizikových aktív do menej rizikových (s menšou smerodajnou odchýlkou) a do hotovosti. Dodatočné ohraničenia možno samozrejme ľu-



Obrázek 4.5: Dosiahnuté výsledky aplikovania lineárneho Mean-CVaR modelu na hladine  $\alpha = 0.95$  s transakčnými nákladmi pre úrovne  $c = 0\%$  (plne),  $c = 0.25\%$  (čiarkovane) a  $c = 1\%$  (bodkočiarkovane).

bovoľne kombinovať a implementovať do naprogramovaných procedúr pre zvolený model portfólia.

#### 4.2.4 Výpočetné vlastnosti optimalizačných metód

V príkladoch generovania efektívnych množín sme optimalizovali portfólia zložené zo siedmich aktív. Z časových dôvodov sme neuvažovali ich väčší počet, keďže rátanie optima v každom bode požadovaného výnosu si vyžaduje istý čas. Aby sme sa však viac priblížili reálnym podmienkam, musíme uvažovať portfólio zložené z väčšieho počtu inštrumentov. Tu nastáva otázka presnosti dosiahnutého výsledku a časového nároku optimalizovania takejto úlohy.

Pre názornejšie ilustrácie budeme tiež pracovať s akciami obsiahnutými v indexe *Standard and Poor's 500 Stock Index* s periódou jeden deň v období 1.1.2004-31.12.2007. S týmito dátami pracujeme rovnako, ako je opísané vyššie pre 7 aktív z indexu *Standard and Poor's 100* s mesačnou periódou.

##### Metóda vnútorného bodu

Metódou vnútorného bodu sme efektívne riešili linearizovaný problém optimálneho portfólia minimalizovaním rizika CVaR. Veľkou výhodou je, že môžeme použiť vysoký počet uvažovaných aktív a scénarov. Použitá metóda pritom vždy konverguje k optimálnemu riešeniu. Pri optimalizovaní portfólia s ôsmymi aktívami (7 rizikových a hotovosť) je simplexová metóda a metóda vnútorného bodu časovo srovnateľná (do 10 sekúnd), metóda vnútorného bodu bola približne o sekundu rýchlejšia.

V prípade akcií z indexu *Standard and Poor's 500* s dennými cenami v 755 historických záznamoch už zreteľne vidno rozdiely. Tabuľka 4.3 zobrazuje čas, ktorý sme k optimalizovaniu potrebovali. Pri voľbe rizika CVaR neparametrickým prístupom, možno uvedený lineárny model optimalizovaný metódou vnútorného bodu odporučiť. Na druhej strane,

exp ret %	2.197	2.055	1.997	1.957	1.288	3.705	1.204	0.25
st dev %	8.88	12.62	9.47	8.5	9.95	15.73	6.57	0
$\omega\%$	Citigroup	Williams	Bank Of NY	Home Dep	Baker	Emc	Coke	Hotovosť
12.	0	0	5.98	35.45	17.41	12.81	8.35	20.
12.66	0	0	12.66	30.48	19.37	15.72	1.77	20.
13.32	0	0	14.98	30.23	20.63	17.11	0	17.05
13.98	0	0	15.71	31.7	21.63	17.94	0	13.01
14.64	0	0	16.44	33.17	22.64	18.78	0	8.98
15.3	0	0	17.17	34.64	23.64	19.61	0	4.95
15.96	0	0	17.9	36.11	24.64	20.44	0	0.91
16.62	0	6.16	10.32	31.57	27.51	24.45	0	0
17.28	0	13.85	0.86	25.23	30.74	29.32	0	0
17.94	0	17.23	0.55	18.09	31.1	33.03	0	0
18.6	0	20.6	0.25	10.96	31.46	36.73	0	0
19.26	0.95	23.44	0	4.59	31.02	40.	0	0
19.92	9.05	22.28	0	4.04	24.64	40.	0	0
20.58	17.15	21.11	0	3.49	18.25	40.	0	0
21.24	25.25	19.95	0	2.94	11.86	40.	0	0
21.9	33.36	18.78	0	2.39	5.47	40.	0	0
22.56	40.	18.64	0	1.36	0	40.	0	0

Tabulka 4.2: Optimálne váhové portfólia lineárneho modelu CVaR pre rôzne hodnoty parametra  $\omega$ -časti portfólia vystavenej riziku.

pre portfólio menších rozmerov je vhodné použiť simplexovú metódu, pretože rozdiely sú zanedbateľné a dostávame tu presné optimum. Zdôrazniť treba, že výsledný čas je ob-

počet aktív	simplexová metóda	metóda vnútorného bodu
100	307 s	144 s
200	759 s	410 s
467	3245 s	1887 s

Tabulka 4.3: Čas optimalizovania lineárneho modelu minimalizovania rizika CVaR s počtom historických záznamov 755.

medzený výpočetnými možnosťami nášho počítača. Zlepšenie metódy vnútorného bodu možno dosiahnuť dodatočným nastavením požadovanej presnosti iterovaných krokov alebo požadovanej presnosti výsledku.

Metódou vnútorného bodu sme tiež hľadali optimálne portfólio Markowitzovho modelu. Napriek tomu, že sme museli doplnením  $2n$  premenných ( $n$  = počet aktív) upraviť ohraňovania transakčných nákladov na lineárne, proces optimalizácie beží veľmi rýchlo. K optimalizovaniu portfólia Markowitzovho modelu sme tiež skúsili použiť metódy popísané v kapitole 3. Metóda vnútorného bodu však vo všetkých prípadoch bola rýchlejšia.

### Stochastické optimalizačné algoritmy

Algoritmy opísané v časti 3 sme použili k optimalizovaniu neparametrickej miery rizika VaR (4.11). Táto funkcia každému prípustnému portfóliu  $\mathbf{x}$  priradí hodnotu pri uvažovanej miere rizika na základe historických dát, alebo generovaných scénarov. K ohodnoteniu

funkcie v bode (portfóliu) je potrebné k zrátniu rizika VaR usporiadať hodnoty výnosov v scénaroch pre dané portfólio. Vzhľadom na veľké množstvo lokálnych extrémov tejto funkcie, potrebujeme v algoritmoch veľké množstvo ohodnotení účelovej funkcie. Náročnosť týchto ohodnotení preto podstatne spomaľuje proces optimalizácie. K optimalizovaniu takejto funkcie v dnešnej dobe nie je zatiaľ preferovaná jedna konkrétna metóda, ale vhodné sú práve stochastické algoritmy. Je potrebné ešte raz zdôrazniť, že hľadanie minima tejto funkcie s uvažovanými dodatočnými ohraničeniami nie je jednoduché, ako sa na prvý pohľad môže zdať.

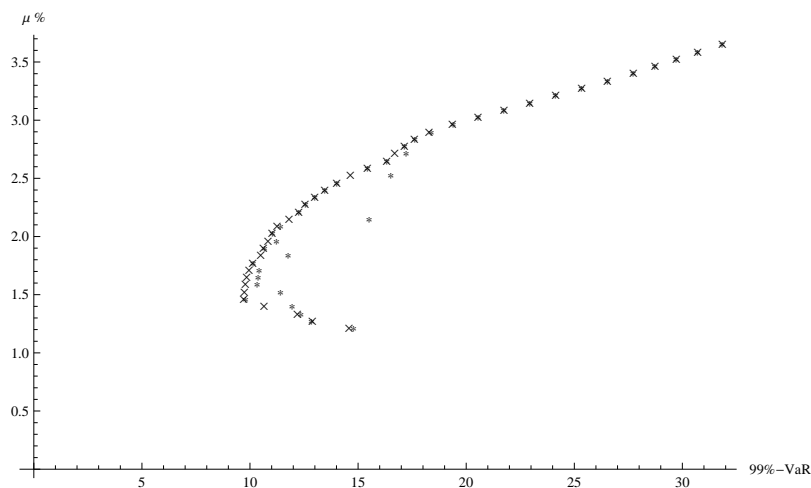
Pri testovaní optimalizačnej procedúry sme zistili, že nahradením požiadavky minimálneho očakávaného výnosu aspoň  $\mu_p$  na požiadavku presného očakávaného výnosu  $\mu_p$  (nahradením rovnosti v ohraničení 4.4), zlepšime optimalizačný proces. Touto úpravou totiž zmenšíme prípustnú množinu a s väčšou pravdepodobnosťou tak neuviazneme na lokálnom minime. Z tohto dôvodu v tejto procedúre uvažujeme presný požadovaný očakávaný výnos. Ako je však na obrázku 4.3 vidieť, aj po tejto úprave sme dosiahli nesprávne globálne minimum. Tu sme sa zameriali na jednotlivé numerické metódy.

Na základe charakteru a testoch Nelder-Meadovej metódy možno povedať, že je nevhodná k optimalizovaniu funkcie rizika VaR. Táto metóda nie je v pravom zmysle globálny optimalizátor. Počiatočný simplex je síce volený náhodne, ale následné iterácie vedú v smere klesania funkcie a tak často uviazne v lokálnom minime. Pri špecifických problémoch dokáže nájsť globálne minimum, aj keď účelová funkcia obsahuje lokálne minima. Pri minimalizácii VaR však skoro vždy v našej úlohe uviazla na lokálnom minime, pričom použitím ostatných metód sme dosiahli lepších výsledkov.

Ako už bolo povedané, vhodné je použiť metódu diferenciálnych evolúcií alebo simulované tavenie. Voľbu interných parametrov vrámci týchto metód (napríklad prechodnú pravdepodobnosť, mutačný faktor) je dobré ponechať na automatickú voľbu, ktorých vhodné nastavenie prevedie softvér. Podstatný vplyv na optimalizačný proces má počet jedincov v generácii u metódy diferenciálnych evolúcií a počet náhodne zvolených počiatočných riešení u metódy simulovaného tavenia. V našom príklade minimalizovania rizika VaR pre portfólio ôsmich aktív sa metóda simulovaného tavenia ukázala lepšia. Je rýchlejšia ako metóda diferenciálnych evolúcií ktorá je náročná na ohodnotenie účelovej funkcie. U metódy simulovaného tavenia sme zvolili 100 náhodných počiatočných riešení a u metódy diferenciálnych evolúcií 100 jedincov populácie. Obrázok 4.6 znázorňuje tieto dosiahnuté riešenia.

Metódu diferenciálnych evolúcií však nemožno všeobecne podceňovať. Jej nevýhodou v našom prípade je náročnosť účelovej funkcie na ohodnotenie. Všeobecne je robustnejšia a má tendenciu neuviaznuť na lokálnom minime. Ako sme už spomínali, DE algoritmus možno naprogramovať paralelne a úlohu optimalizovať na viacerých procesoroch. Tým dosiahneme jej výrazne zefektívnenie.

Venujme sa ešte jednej dôležitej skutočnosti pri minimalizovaní neparametrickej miery VaR. Ako sme už vraveli, táto funkcia má značný počet lokálnych extrémov. Toto môže prispieť k tomu, že pri optimalizovaní dostaneme relatívne rozdielne portfólia vzhľadom k získanému minimu rizika VaR. Z tohto dôvodu sme spustili optimalizačnú procedúru pri rovnakých požadovaných výnosoch 1,53182% na hladine  $\alpha = 0.99$  a len pri rôznych hodnotách nastavenia pseudogenerátora náhodných čísel (v systéme Mathematica nastavenie `RandomSeed`). Tabuľka 4.4 znázorňuje tieto dosiahnuté výsledky použitím metódy simulovaného tavenia so 100 štartovacími riešeniami. Je vidieť, že zloženie portfólia je citlivé na dosiahnuté minimum rizika VaR. V našom prípade k tejto skutočnosti prispelo použí-



Obrázek 4.6: Porovnanie metód SA a DE pri minimalizovaní neparametrickej miery 0,99-VaR. Znak „\*“ pre DE a „x“ pre SA.

RandomSeed	0.99-VaR (%)	$x_1$ (%)	$x_2$ (%)	$x_3$ (%)	$x_4$ (%)	$x_5$ (%)	$x_6$ (%)	$x_7$ (%)
1	9.7554	2.43398	5.63682	4.68669	19.7655	21.5667	2.0701	43.8402
2	9.72992	2.70491	7.91697	4.31186	20.5627	19.001	1.15138	44.3512
3	9.76878	5.19702	8.58604	0.667996	22.4297	18.2722	0.552445	44.2945
4	9.72834	3.18432	7.99021	6.4625	20.5225	18.5125	0.282077	43.0459
5	9.75271	3.931	7.48411	4.69844	21.0341	19.1526	0.542052	43.1577

Tabulka 4.4: Percentuálne váhy dosiahnutých optimálnych portfólií minimalizováním neparametrickej miery VaR pre rôzne nastavenia generátora náhodných čísel RandomSeed. Očakávaná miera výnosu je 1,53182%.

tie relatívne malého počtu historických záznamov (204 mesačných periód). Je preto lepšie použiť ich väčší počet, alebo generovať dostatočný počet scénarov. Neparametrická miera VaR sa pri ich vyššom počte „vyhľadáje“ a jej graf sa podobá grafu parametrickej miery VaR, ktorá je v prípade normálnych rozdelení konvexnou funkciou. So zvýšeným počtom scénarov (historických záznamov) však narastá obtiažnosť ohodnotenia funkčnej hodnoty VaR. Niektorí správcovia portfólií preto používajú neparametrickú mieru CVaR, ktorá je konvexnou funkciou. A tiež preto, že portfólio s nízkym rizikom CVaR, musí mať aj nízke riziko VaR (vlastnosť  $CVaR_\alpha(\mathbf{x}) \geq VaR_\alpha(\mathbf{x})$ ).



# Kapitola 5

## Záver

V tejto diplomovej práci sme sa venovali vybraným metódam matematického programovania a optimalizácie. Zameriali sme sa na matematické modely v súvislosti s problémom výberu optimálneho portfólia. Pre kritérium optimality sme zvolili očakávaný výnos a riziko v podobe mier smerodajnej odchýlky, strednej absolútnej odchýlky a miery rizika VaR a CVaR. K meraniu rizika sme pristupovali parametricky, alebo neparametricky použitím historických dát. Použili sme pritom matematické programy lineárneho, kvadratického a všeobecne nelineárneho programovania. Oboznámili sme sa so základnou technikou na riešenie lineárnych úloh, simplexovou metódou a efektívnejšou numerickou metódou vnútorného bodu pre lineárne programovanie. Modifikovanú metódu vnútorného bodu sme tiež predstavili k riešeniu problému konvexného kvadratického programovania. Pre všeobecne nelineárne úlohy sme zvolili moderné stochastické optimalizačné algoritmy, venovali sme sa Nelder-Meadovej metóde, simulovanému taveniu a metóde diferenciálnych evolúcií.

Markowitzov model s rizikom smerodajnej odchýlky sme implementovali kvadratickým programom a prakticky optimalizovali metódou vnútorného bodu. Model s mierou rizika strednej absolútnej odchýlky sme pomocou historických dát previedli na lineárny problém. Podobne, model s rizikom CVaR v ohraničnej sme pomocou prezentovaného teoretického aparátu linearizovali a efektívne riešili technikami pre lineárne programovanie. Metóda vnútorného bodu, sa hlavne pri väčšom počte premenných ukázala efektívnejšia.

K miere rizika VaR sme zvolili neparametrický prístup. Táto miera, počítaná na základe historických dát (alebo scénarov) je nekonvexnou funkciou s množstvom lokálnych extrémov. Minimalizovali sme ju použitím stochastických optimalizačných metód. Vhodná sa ukázala metóda diferenciálnych evolúcií a simulovaného tavenia, pričom druhá menovaná dosiahla lepších výsledkov pre portfólio ôsmich aktív. Ukázali sme tiež, že dosiahnuté portfólia pri minimalizovaní neparametrickej miery VaR sú citlivé na relatívne malé zmeny dosiahnutých hodnôt rizika VaR. Pozornosť je preto potrebné venovať dodatočnému nastaveniu použitých metód a počtu a spôsobu použitia historických dát alebo scénarov. Do problémov optimálnych portfólií sme tiež zahrnuli dodatočné ohraničenia ako transakčné náklady, ohraničenia hodnôt a na príkladoch ukázali ich vplyv na voľbu optimálneho portfólia.

Všetky optimalizačné procedúry na riešenie popísaných modelov s možnosťou implementácie reálnych ohraničení a dát, sme naprogramovali prostredníctvom systému Mathematica a na prezentované príklady použili historické ceny akcií obsiahnutých v indexoch Standard and Poor's.

# Literatura

- [1] Baeck, T., Fogel, D., Michalewicz, Z. (1997): *Handbook of Evolutionary Computation*, Oxford University Press, section C 5.2.
- [2] Cornuejols, G., Tutuncu, R. (2005): *Optimization Methods in Finance*, Carnegie Mellon University, Pittsburgh.
- [3] Dupačová, J., Hurt, J., Štěpán, J. (2002): *Stochastic Modeling in Economics and Finance*, Kluwer Academic Publishers, Dordrecht.
- [4] Dupačová, J., Lachout, P. (2008): *Úvod do optimalizace*, <http://www.karlin.mff.cuni.cz/~lachout/>.
- [5] Hurt, J. (2008): *Risk Measures in Finance, International Mathematica User Conference 2008*, <http://library.wolfram.com/infocenter/conferences/7230/>.
- [6] Krokhmal, P., Palmquist, J., Uryasev, S. (2001): *Portfolio Optimization with Conditional Value-at-Risk Objective and Constraints*, <http://www.ise.ufl.edu/uryasev>.
- [7] Lenstra, J. K., Rinnooy K., A. H. G., Schrijver, A. (1991): *History of Mathematical Programming: A Collection of Personal Reminiscences*, Elsevier Science Publishers B.V., Amsterdam.
- [8] Locatelli, M. (2000): *Simulated Annealing Algorithms for Continuous Global Optimization*, [www.di.unito.it/~locatell/sa.ps](http://www.di.unito.it/~locatell/sa.ps).
- [9] Morgan, J. P., Reuters (1996): *Risk Metrics-Technical Document. 4th ed.*, Morgan Guaranty Trust Company, New York.
- [10] Nelder, J. A., Mead, R. (1965): *A Simplex Method for Function Minimization*, The Computer Journal **7**(4), 308-313.
- [11] Pflug, G. Ch. (2000): *Some Remarks on the Value-at-Risk and the Conditional Value-at-Risk*, Kluwer Academic Publishers.
- [12] Rockafellar, R. T., Uryasev, S. (2000): *Optimization of Conditional Value-at-Risk*, Journal of Risk **2**(3), 21-44.
- [13] Storn, R., Price, K. (1997): *Differential Evolution - A Simple and Efficient Heuristic for global Optimization over Continuous Spaces*, Journal of Global Optimization **11**(4), 341-359.
- [14] Vanderbei, R. (2001): *Linear Programming: Foundations and Extensions*, Springer-Verlag, New York.