

FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY
UNIVERZITY KOMENSKÉHO
v Bratislave

Ekonomická a finančná matematika



DIPLOMOVÁ PRÁCA

Metódy vnútorného bodu v kvadratických úlohách
optimalizácie portfólia

Autor: Milan Výškrabka

Vedúci diplomovej práce: Doc. RNDr. Margaréta Halická, CSc.

Bratislava 2003

Obsah

Úvod	1
1. Markowitzov model portfólia	3
1.1 Portfólio	3
1.2 Vytvorenie portfólia	5
1.3 Markowitzov model	8
1.4 Transakčné náklady	11
2. Kvadratické programovanie	15
2.1 Primárna úloha	15
2.2 Podmienky optimality	16
2.3 Duálna úloha	18
3. Metódy vnútorného bodu	19
3.1 Centrálna trajektória	19
3.2 Bariérový problém	20
3.3 Podmienky optimality v bariérovom probléme	21
3.4 Nepripustná primárno-duálna metóda sledujúca centrálnu trajektóriu	23
3.4.1 Výpočet smerov	24
3.4.1.1 Newtonova metóda	26
3.4.2 Výpočet dĺžky kroku	27
3.4.3 Odhad parametra μ	29
3.4.4 Štart a koniec algoritmu	31
3.4.5 Zhrnutie	32
4. Numerický experiment	34
4.1 LOQO	34
4.2 Úloha s fixnými transakčnými nákladmi	39
Záver	43
Literatúra	44

Úvod

Vytvorenie portfólia, ktoré bude dosahovať solídne výnosy počas celej doby nie je len problém vybrať vhodné aktíva do portfólia, ale z už vybraných aktív správne zvoliť ich váhy v portfóliu. Vhodné zastúpenie jednotlivých aktív nemožno určiť len prirodzenou intuíciou. Úspech v investovaní môžeme dosiahnuť, ak budeme precízne analyzovať aktíva a následne optimalizovať portfólio. Optimalizácia portfólia je proces analýzy aktív a riadenia aktív v portfóliu tak, aby výnos portfólia bol pri danej hodnote rizika najvyšší. Praktická realizácia spočíva v modelovaní portfólia ako optimalizačného matematického modelu. Preto je záujmom matematikov hľadať metódy ako efektívne takéto úlohy riešiť. Medzi nové a populárne metódy sa jednoznačne zaradili metódy vnútorného bodu. Práve jednou z týchto metód sa zaoberá táto diplomová práca.

Cieľom teoretickej časti práce je na modeli optimálneho portfólia popísať algoritmus primárno-duálnej metódy vnútorného bodu. Cieľom praktickej časti bolo získať prehľad o dostupnosti softvéru na internete, v ktorom je implementovaná spomenutá metóda a pre rôzne dimenzie úloh optimalizácie portfólia otestovať funkčnosť a rýchlosť vybraného softvéru. Počítané úlohy boli modely optimalizácie portfólia s fixnými transakčnými nákladmi ako aj s variabilnými transakčnými nákladmi. Výsledky sú zhrnuté v závere poslednej kapitoly.

Prvá kapitola sa venuje teoretickým poznatkom Markowitzovho modelu portfólia. Pred samotným vytvorením portfólia potrebujeme poznať očakávané výnosy a volatilitu akcií. V prvej časti uvádzame možnosť výpočtu očakávaných návratností. Na príklade dvoch aktív v portfóliu si ukážeme ako sa tvorí portfólio. Tento príklad potom zovšeobecníme na ľubovoľný počet aktív v portfóliu. Výsledkom je model portfólia, v ktorom sa hľadá vhodná kombinácia výnosu a rizika pri ohraničeniach, ktoré obmedzujú držanie aktív. Záver kapitoly je venovaný reálnej situácii na trhu, kde každý obchod s akciami je spoplatnený. Markowitzov model rozšírime o transakčné náklady. Model odvodený v tejto kapitole je úlohou kvadratického programovania. Zdrojom informácií boli [1], [5], [10].

V druhej kapitole popíšeme vlastnosti úlohy kvadratického programovania. Odvodíme duálnu úlohu k primárnej úlohe a podmienky optimality k obom úlohám. Dosiahnuté poznatky využijeme pri odvodení algoritmu metódy vnútorného bodu. Kapitola bola spracovaná podľa [2].

Cieľom tretej kapitoly je podrobne popísať algoritmus, ktorý rieši úlohu sformulovanú v prvej kapitole v polynomiálnom čase. Úlohu z prvej kapitoly upravíme na tzv. bariérovú úlohu. Podmienky optimality odvodené podľa kapitoly 2 využijeme v takejto úlohe na získanie algoritmu, ktorý v závere kapitoly uvedieme v prehľadnej schéme.

Cieľom poslednej kapitoly je vyskúšať softvér, v ktorom je implementovaný algoritmus popísaný v tretej kapitole, na úlohe s transakčnými nákladmi z prvej kapitoly. Na začiatku kapitoly sú popísané charakteristiky softvéru. Druhá polovica kapitoly sa venuje konkrétnym numerickým experimentom. Vďaka Jaroslave Hlouškovej z Institut für Höhere Studien Viedeň, ktorá poskytla všetky dáta pre úlohu s fixnými transakčnými nákladmi, bol softvér odskúšaný na úlohách rôznych rozmerov. Najmenšia úloha s fixnými transakčnými nákladmi obsahovala 100 aktív. Najväčšia obsahovala 500 aktív.

MARKOWITZOV MODEL PORTFÓLIA

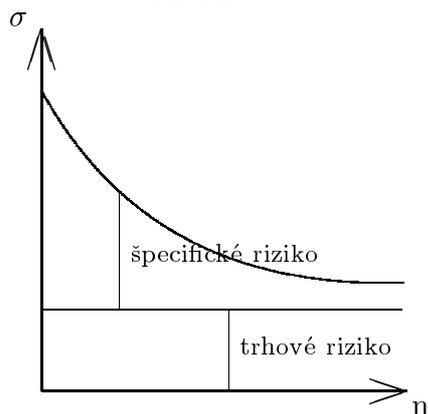
1.1 Portfólio

Vo svete financií používame pojem portfólio na označenie súboru aktív, pričom aktívami môžu byť akcie, dlhopisy alebo iné inštrumenty obchodované na finančnom trhu. Zmysel investovania na finančných trhoch je v zarábaní peňazí. Každý investor sa snaží svoje prostriedky zhodnotiť čo najviac. Táto práca sa zaoberá portfóliom obsahujúcim iba akcie (rizikové aktíva).

Vytvorenie portfólia pozostáva z dvoch aspektov. Prvým je výber konkrétnych aktív do portfólia. Trh v súčasnosti ponúka obrovský výber. Druhým aspektom je určenie pomerného zastúpenia vybraných aktív v portfóliu, s čím súvisí optimalizácia.

Každá investícia je však spojená s rizikom, ktorým je neistota z návratu investície. Spravidla platí, že čím je očakávaný výnos vyšší, tým je vyššie aj riziko. Uvažujme portfólio s dvoma aktívami. Ak napríklad zvýšená cena ropy negatívne zasiahne jednu firmu, druhej to môže pomôcť, pretože poskytuje produkty, ktoré sú vhodnou alternatívou produktov spoločnosti závislej na rope. Dve takéto aktíva

OBR. 1.

Vzťah medzi rizikom σ a počtom akcií n v portfóliu.

v portfóliu môžu dať spolu istejší výnos ako keby sme držali len jedno aktívum. Preto je vhodné do portfólia zahrnúť viac aktív, čím môžeme riziko diverzifikovať.

Pridávaním akcií donekonečna sa však riziko nedá eliminovať úplne. Ak sú všetky zdroje rizika na sebe nezávislé a ak portfólio obsahuje veľa aktív, riziko sa dá do značnej miery znížiť (obr.1). Predpokladáme, že investor sa už rozhodol, ktoré akcie chce držať v portfóliu. My sa budeme zaoberať druhým aspektom tvorby portfólia. V portfóliu, ktoré je tvorené iba z akcií firiem (rizikové aktíva), rozlišujeme dva druhy zdrojov rizika. Jeden zdroj tvorí trhové riziko (systematické, nediverzifikovateľné), ktoré vzniká z neistôt vyplývajúcich z úrokových sadzieb, inflácie, výmenných kurzov a iných neovplyvniteľných ekonomických faktorov. Tieto faktory majú rovnaký vplyv na všetky firmy (na ich hospodárenie, a teda aj na hodnotu ich akcií). Pridaním ďalšej akcie do portfólia toto riziko ostáva v plnej výške. Nemôžeme ho ovplyvniť. Druhým je špecifické riziko (nesystematické, diverzifikovateľné) plynúce z rizika firmy. Diverzifikácia rizika je proces pridávania akcií do portfólia za účelom zníženia špecifického rizika. Čím väčší počet akcií portfólio obsahuje, tým je možné lepšie rozložiť riziko. Trhové riziko diverzifikáciou znížené byť nemôže. Vzťah medzi očakávaným výnosom a rizikom portfólia ako prvý naformuloval Harry Markowitz, ktorý za svoju prácu z roku 1952 získal Nobelovu cenu. Markowitz definuje riziko portfólia ako varianciu výnosu portfólia.

Jeden zo spôsobov ako získať očakávaný výnos portfólia je odhadnúť ho z historických dát. Uvažujme časový horizont dĺžky T (napr. pri mesačnej periodicite bude pre jeden rok $T=12$). Označme $r_i(t)$ výnos akcie i v čase t , kde $i = 1, \dots, n$, $t = 1, \dots, T$. Keďže r_i je náhodná veličina, používame očakávaný výnos (stredná hodnota) $E(r_i)$,

$$E(r_i) = \frac{1}{T} \sum_{t=1}^T r_i(t). \quad (1.1)$$

Takýto odhad očakávaného výnosu však nie je veľmi reálny. Zrejme udalosti, ktoré sa odohrali ku koncu sledovaného obdobia ($t \rightarrow T$) vplývajú na súčasnosť väčšou váhou ako udalosti na začiatku sledovaného obdobia ($t \rightarrow 0$). Toto tvrdenie získava na dôležitosť s rastúcim intervalom sledovania vývoja akcií T . Formula (1.1) priraduje každému časovému okamihu rovnakú váhu. Presnejšie odhady výnosu získame diskontovaním výrazu (1.1)

$$E(r_i) = \frac{\sum_{t=1}^T p^{T-t} r_i(t)}{\sum_{t=1}^T p^{T-t}}, \quad (1.2)$$

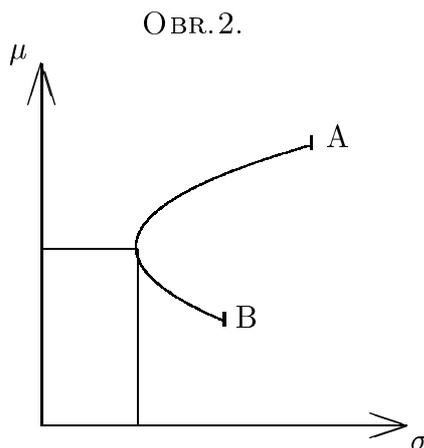
kde $p \in (0, 1)$ je diskontný faktor. Pre $p = 1$ získame výraz (1.1). Pre $p < 1$ dávame väčší dôraz na nedávne informácie. Čím je p menšie, tým súčasnosť ($t=T$) získava väčšiu váhu oproti minulosti. Takto získame reálnejší odhad očakávaného výnosu. Praktické experimenty ukázali, že presnejšie odhady získame, ak urobíme nasledovnú úpravu

$$E(r_i) = \exp \left(\frac{\sum_{t=1}^T p^{T-t} \ln r_i(t)}{\sum_{t=1}^T p^{T-t}} \right). \quad (1.3)$$

1.2 Vytvorenie portfólia

Ilustrujme najprv na príklade dvoch rizikových aktív A a B ako sa vytvorí portfólio. V ďalšej časti potom rozšírime výber na $n > 2$ rizikových aktív.

Na obrázku 2 je znázornený výnos a riziko akcií A a B. Krivka, ktorá spája body A a B predstavuje možné portfólia zostavené z akcií A a B. Táto krivka musí prechádzať bodmi A a B, pretože portfólio, ktoré obsahuje iba jeden druh aktíva (A alebo B) je tiež možné zostrojiť a teda musí ležať na krivke. Cieľom je uvedomiť si na tomto jednoduchom príklade ako sa konštruuje portfólio. Vytvorenie portfólia s viacerými aktívami vykazuje analógiu s týmto jednoduchým príkladom.



Krivka spájajúca body A a B znázorňuje minimálne riziko σ_P pre ľubovольnú úroveň výnosu μ_P .

Nech má akcia A v portfóliu váhu x_1 . Potom zvyšok investície budeme držať v akciách B, ktoré budú mať váhu x_2 . Teda $x_2 = 1 - x_1$. Portfólio bude mať očakávaný výnos $E(r_P)$ daný ako

$$E(r_P) = x_1 E(r_1) + x_2 E(r_2), \quad (1.4)$$

kde $E(r_i)$, $i = 1, 2$ je očakávaná návratnosť aktíva A, resp. B. $E(r_i)$ môže byť získané napr. z (1.3).

Ak $\sigma_{ij} = cov(r_i, r_j)$, $i = 1, 2$ je kovariancia náhodných premenných r_i a r_j , tak $\sigma_{ij} = \sigma_{ji}$. Teda variancia výnosu portfólia σ_P^2 je

$$\begin{aligned} \sigma_P^2 &= x_1^2 \sigma_{11} + x_2^2 \sigma_{22} + 2x_1 x_2 \sigma_{12} \\ &= x_1^2 \sigma_{11} + (1 - x_1)^2 \sigma_{22} + 2x_1(1 - x_1) \sigma_{12}. \end{aligned} \quad (1.5)$$

Teda variancia výnosu portfólia je váženým priemerom kovariancií. Kovarianciu dvoch náhodných veličín r_1, r_2 môžeme zapísať ako

$$\sigma_{12} = cov(r_1, r_2) = \rho_{12}\sigma_1\sigma_2,$$

kde $\sigma_1^2 = cov(r_1, r_1)$, $\sigma_2^2 = cov(r_2, r_2)$ a $\rho_{12} \in \langle -1, 1 \rangle$ je korelačný koeficient. Korelačný koeficient vyjadruje mieru závislosti r_1 od r_2 . Označme $\sigma_{11} \equiv \sigma_1^2$, $\sigma_{22} \equiv \sigma_2^2$, $\sigma_{12} \equiv \rho_{12}\sigma_1\sigma_2$. Uvažujme extrémálne hodnoty $\rho_{12} = \pm 1$, potom

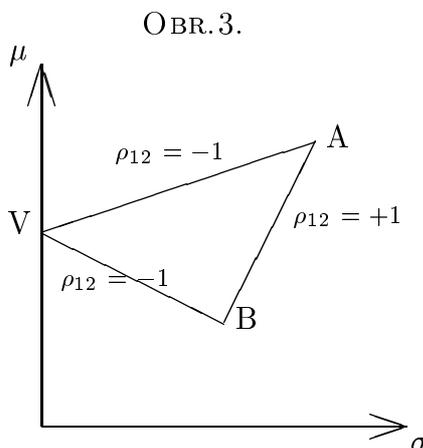
$$\begin{aligned}\sigma_P^2 &= x_1^2\sigma_1^2 \pm 2x_1(1-x_1)\sigma_1\sigma_2 + (1-x_1)^2\sigma_2^2 \\ &= (x_1\sigma_1 \pm (1-x_1)\sigma_2)^2, \\ \sigma_P &= |x_1\sigma_1 \pm (1-x_1)\sigma_2|.\end{aligned}$$

Štandardná odchýlka σ_P musí byť nezáporná. Preto

$$\rho_{12} = +1 \implies \sigma_P = x_1\sigma_1 + (1-x_1)\sigma_2$$

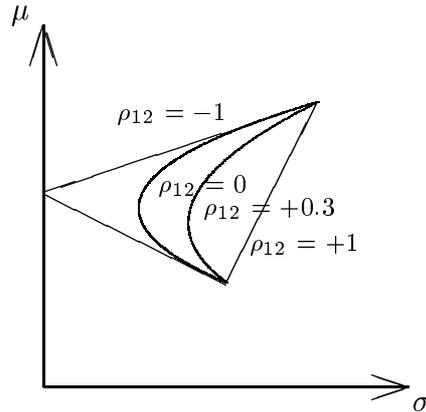
$$\rho_{12} = -1 \implies \sigma_P = \begin{cases} (x_1\sigma_1 - (1-x_1)\sigma_2) \geq 0, & \text{pre } x_1 \geq \frac{\sigma_2}{\sigma_1 + \sigma_2} \\ -(x_1\sigma_1 - (1-x_1)\sigma_2) > 0, & \text{pre } x_1 < \frac{\sigma_2}{\sigma_1 + \sigma_2}. \end{cases} \quad (1.6)$$

Výrazy (1.6) sú lineárne a vymedzujú priestor pre možné portfólia zostavené z aktív A a B (obr.3).



Hranica minimálnej variancie pre extrémálne hodnoty korelačného koeficientu ρ_{12} .

OBR. 4.

Hranice minimálnej variance pre rôzne hodnoty korelačného koeficientu ρ_{12} .

V závislosti od korelačného koeficientu ρ_{12} budú efektívne portfólia ležať niekde v priestore určenom bodmi ABV. Príkladom sú krivky na obr.4. Riziko, teda variácia výnosu, portfólia je podľa (1.5) najväčšie, ak $\rho_{12} = +1$. Preto nie je vhodné zahrnúť do portfólia aktíva, ktorých korelačný koeficient dosahuje hodnotu rovnú jednej alebo blízku tejto hodnote. Naopak, variácia je najnižšia, keď $\rho_{12} = -1$. Najmenšie riziko aké môžeme kombináciou aktív A a B dosiahnuť pri váhe akcií A x_1^* získame položením $\frac{\partial \sigma_P^2}{\partial x_1} = 0$. Úpravou výrazu dostaneme

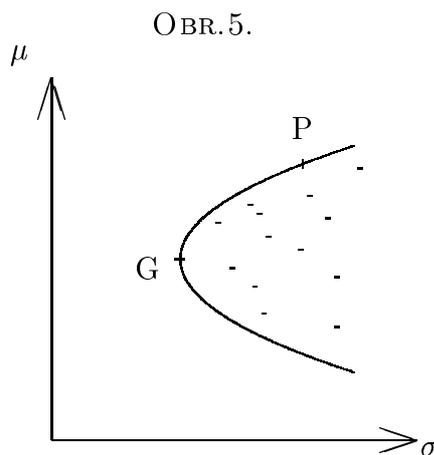
$$x_1^* = \frac{\sigma_2^2 - \rho_{12}\sigma_1\sigma_2}{\sigma_1^2 + \sigma_2^2 - 2\rho_{12}\sigma_1\sigma_2}.$$

V závislosti od korelačného koeficientu ρ_{12} dostávame rôzne váhy.

1.3 Markowitzov model

Teraz zovšeobecníme prípad výberu dvoch rizikových aktív na výber n rizikových aktív. Výsledkom je nájdenie množiny najlepších kombinácií rizika a návratnosti aktív. Portfólia, ktoré dosahujú daný výnos a riziko ležia na krivke, ktorá

sa nazýva *hranica minimálnej variancie*. Portfólio z tejto hranice má najnižšie možné riziko, ktoré môže byť dosiahnuté pri danej očakávanej návratnosti. Všetky individuálne aktíva ležia na pravej strane od tejto hranice. Ak je zakázaný krátky predaj, niektoré aktíva môžu ležať aj na hranici (zakázaný krátky predaj znamená, že nemôžeme predať aktívum, ktoré nedržíme vo svojom portfóliu) Napr. akcia s najvyšším výnosom musí ležať na hranici, pretože je to aktívum, ktoré ako jediné prináša ten požadovaný výnos (obr.5).



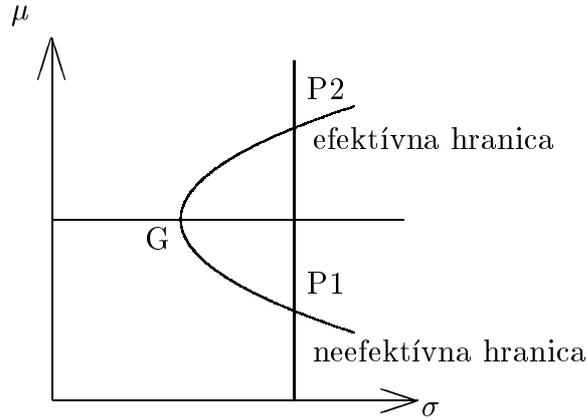
Hranica minimálnej variancie a jednotlivé aktíva.

Bod G ležiaci na vrchole paraboly je globálnym variančným minimom. Všetky body ležiace na hranici a nad bodom G sú najlepšími možnými kombináciami výnosu a rizika portfólia a tvoria tzv. *efektívnu hranicu*. Pre ľubovoľné portfólio ležiace na hranici pod bodom G existuje portfólio s rovnakým rizikom, ale vyšším výnosom. Preto spodná časť paraboly tvorí tzv. *neefektívnu hranicu* (obr.6).

Teraz sa budeme zaoberať problematikou hľadania konkrétneho portfólia P z hranice minimálnej variancie.

Predpokladáme, že investor plánuje investovať na jeden časový horizont, napr. jeden rok. Pre *i* – te aktívum ($i = 1, \dots, n$) urobíme odhad očakávanej návratnosti $E(r_i)$ a kovariancie $cov(r_i, r_j)$ za zvolené obdobie. Potom očakávaný výnos

OBR. 6.



Portfólio P2 na efektívnej hranici má pri rovnakom riziku vyšší výnos ako portfólio P1 na neefektívnej hranici.

portfólia μ_P a variancia σ_P sú

$$\mu_P = E(r_P) = \sum_{i=1}^n x_i E(r_i), \quad (1.7)$$

$$\sigma_P^2 = \sum_{i=1}^n x_i^2 \sigma_i^2 + \sum_{\substack{i=1 \\ i \neq j}}^n \sum_{j=1}^n x_i x_j \text{cov}(r_i, r_j), \quad (1.8)$$

ktoré vyplývajú z (1.4) a (1.5).

Nech $\mu = (\mu_1, \dots, \mu_n)^T$ označuje vektor očakávaných výnosov, kde $\mu_i \equiv E(r_i)$, $i = 1, \dots, n$ a $C \equiv [\sigma_{ij}]$, $i, j = 1, \dots, n$ je štvorcová matica kovariancií,

kde $\sigma_{ij} = \text{cov}(r_i, r_j)$. C je potom symetrická a kladne semidefinitná. Ak $x_i \geq 0$ je časť, ktorú chceme investovať do aktíva i , tak $x \equiv (x_1, \dots, x_n)^T$ sa nazýva portfólio.

Na portfólio sa vzťahujú mnohé ohraničenia, ktoré môžu byť: zakázané krátke pozície ($x \geq 0$), rozpočtové ohraničenie ($l^T x = 1$, $l = (1, \dots, 1)^T$) a mnohé ďalšie ako napr. limity na držanie aktív z jedného odvetvia, požadovaný dividendový výnos alebo ohraničenia plynúce zo zákona atď.

Potom úlohu optimalizovania portfólia môžeme sformulovať ako úlohu kvadratického programovania, kde minimalizujeme riziko pri zadanej hodnote očakávaného výnosu μ_P s rozpočtovým ohraňením a zakázaným krátkym predajom.

$$\min_x \left\{ \frac{1}{2} x^T C x \mid \mu^T x = \mu_P, l^T x = 1, x \geq 0 \right\}. \quad (1.9)$$

Analogicky sa na problém hľadania optimálneho portfólia dá pozrieť ako na úlohu hľadania maximálneho očakávaného výnosu pri danej úrovni rizika σ_P^2 , ktoré sme ochotní podstúpiť. Tiež uvažujeme rozpočtové ohraničenie a zakázaný krátky predaj

$$\max_x \{ \mu^T x \mid x^T C x \leq \sigma_P^2, l^T x = 1, x \geq 0 \}. \quad (1.10)$$

Úlohy (1.9) a (1.10) môžeme zapísať ako (1.11), ktorej riešením v závislosti od parametra $t \geq 0$ získame portfólio z hranice minimálnej variancie

$$\min_x \{ -t\mu^T x + \frac{1}{2}x^T C x \mid l^T x = 1, x \geq 0 \}. \quad (1.11)$$

Parameter t môžeme chápať ako parameter tolerancie rizika. Ak $t = 0$, úloha (1.11) sa zúži na úlohu minimalizovania rizika. V tomto prípade hovoríme, že investor je averzný k riziku. Ak t je veľké, výnos (lineárny člen) v účelovej funkcii zohráva v porovnaní s rizikom veľkú úlohu. Matica C (matica kovariancií) je kladne semidefinitná, preto sú úlohy (1.9), (1.10) a (1.11) úlohami konvexného programovania.

1.4 Transakčné náklady

Ceny akcií vykazujú pomerne vysokú volatilitu. Ich hodnoty sa denne menia. Tým sa menia aj odhadované výnosy, resp. variancie, čím sa nami držané portfólio odchyľuje od optimálneho stavu. Aby sme zachovali pri požadovanom očakávanom výnose minimálne riziko, musíme portfólio prerovnávať, teda kupovať resp. predávať akcie tak, aby sa stav aktív vrátil do optimálneho stavu. V reálnom svete však nemôžeme kúpiť, resp. predať aktíva bez dodatočných nákladov, tzv. *transakčných nákladov*. Ak náklady na zmenu v držaní aktíva sú vysoké, nemusí sa zmena oplatiť. Ale transakčné náklady sú v niektorých prípadoch zanedbateľné

oproti potenciálnemu zisku zo zmeny v držaní aktíva a vtedy obchod uskutočníme. Nech x^+ a $x^- \in R^n$ sú vektory nakúpených, resp. predaných aktív. Nové portfólio x musí vo vzťahu k existujúcemu portfóliu \hat{x} spĺňať

$$x = \hat{x} + x^+ - x^-, \quad x^+ \geq 0, \quad x^- \geq 0. \quad (1.12)$$

Transakčné náklady spojené s nákupom resp. predajom aktív môžu byť fixné alebo variabilné. Fixné náklady sú rovnaké bez ohľadu na objem transakcie. Variabilné transakčné náklady sa menia s veľkosťou transakcie. Väčšie obchody majú zvyčajne väčší dopad na zmenu ceny aktíva, preto táto zmena je bránená väčšími nákladmi za obchod.

Pri fixných nákladoch je možné použiť lineárnu funkciu na spoplatnenie transakcie. Pre i -te aktívum ($i = 1, \dots, n$) predpokladáme, že percentuálne náklady na kúpu i -teho aktíva sú p_i a náklady na predaj i -teho aktíva sú q_i . Celkové náklady na zmenu portfólia sú

$$p^T x^+ + q^T x^-, \quad (1.13)$$

kde $p = (p_1, \dots, p_n)^T$, $q = (q_1, \dots, q_n)^T$. Jeden spôsob ako zahrnúť transakčné náklady do modelu (1.11) je znížiť očakávaný výnos o tieto náklady. Dostávame

$$\min_{x, x^+, x^-} \left(-t(\mu^T x - p^T x^+ - q^T x^-) + \frac{1}{2} x^T C x \right) \quad (1.14)$$

pri podmienkach

$$l^T x = 1$$

$$x - x^+ + x^- = \hat{x}$$

$$x \geq 0, \quad x^+ \geq 0, \quad x^- \geq 0,$$

kde $\hat{x} \in R^n$ je pôvodné portfólio. Úloha (1.14) je úlohou kvadratického programovania s $3n$ neznámymi (x, x^+, x^-) a $4n + 1$ lineárnymi ohraničeniami. Túto úlohu môžeme všeobecne napísať (pre konkrétnu hodnotu parametra t , rôzne ohraničenia na držanie aktív)

$$\min_{\tilde{x}} \left(c^T \tilde{x} + \frac{1}{2} \tilde{x}^T \tilde{C} \tilde{x} \right)$$

pri podmienkach

$$A \tilde{x} \leq b \tag{1.15}$$

$$\tilde{x} \geq 0,$$

kde $\tilde{x} = (x, x^+, x^-)^T$ je vektor neznámych, $c = t(-\mu, p, q)^T \in R^{3n}$, $\tilde{C} = \begin{pmatrix} C & 0 \\ 0 & 0 \end{pmatrix}$ je $(3n, 3n)$ matica, A je $(m, 3n)$ matica ohraňení, $b \in R^m$ je vektor pravých strán. V ďalšej časti sa budeme venovať úlohe s fixnými transakčnými nákladmi, teda úlohe (1.14) resp. (1.15).

Variabilné transakčné náklady rozdeľujú rozsah obchodovania každého aktíva na niekoľko intervalov. Každému intervalu je priradená určitá výška transakčných nákladov. Takže malé obchody môžu mať malé náklady na transakcie, zatiaľ čo za veľké obchody investor zaplatí veľké transakčné náklady. Aj pri tomto spôsobe spoplatnenia transakcií platí (1.12), kde

$$x^+ = x_1^+ + x_2^+ + \dots + x_k^+,$$

$$x^- = x_1^- + x_2^- + \dots + x_k^-,$$

t.j. x^+ a x^- sú vyjadrené ako suma kupovaných resp. predávaných aktív v každom intervale, kde $0 \leq x_i^- \leq d_i$, $0 \leq x_i^+ \leq e_i$, $i = 1, \dots, k$. Nákupné transakčné náklady pre prvý interval $(\hat{x}, \hat{x} + e_1)$ sú $p_1^T x_1^+$. Ak obchod pokračuje do druhého intervalu $(\hat{x} + e_1, \hat{x} + e_1 + e_2)$ transakčné náklady sú $p_1^T x_1^+ + p_2^T x_2^+$ atď.

Všeobecne celkové transakčné náklady sú

$$p_1^T x_1^+ + \dots + p_k^T x_k^+ + q_1^T x_1^- + \dots + q_k^T x_k^-.$$

Tento model určuje horné a dolné ohraňenia na držanie aktíva v portfóliu. Hranice sú dané: $\hat{x} + e_1 + \dots + e_k$ resp. $\hat{x} - d_1 - \dots - d_k$. Model (1.14) sa zmenením

fixných nákladov na transakčné náklady zmení nasledovne:

$$\begin{aligned}
 \min & \left(-t(\mu^T x - p_1^T x_1^+ - \dots - p_k^T x_k^+ - q_1^T x_1^- - \dots - q_k^T x_k^-) - \frac{1}{2}x^T Cx \right) \\
 & l^T x = 1 \\
 & x - x_1^+ - \dots - x_k^+ + x_1^- + \dots + x_k^- = \hat{x} \\
 & 0 \leq x_i^- \leq d_i, \quad i = 1, \dots, k \\
 & 0 \leq x_i^+ \leq e_i, \quad i = 1, \dots, k,
 \end{aligned} \tag{1.16}$$

kde $d, e \in R^n$ sú vektory horných ohraničení na nákup, resp. predaj. V realite je bežné, že portfólio obsahuje tisíc aktív. Po zarátaní konštantných transakčných nákladov máme 3000 neznámych, čo je už veľmi rozsiahla úloha. Pre bežné algoritmy, ktoré úlohy kvadratického programovania riešia v exponenciálnom čase, je to úloha časovo veľmi náročná. Preto sa žiada hľadať nové metódy, ktoré by uvedený problém riešili efektívnejšie. V súčasnosti sa východisko ukazuje v metódach vnútorného bodu, ktoré vykazujú polynomiálnu zložitosť.

KVADRATICKÉ PROGRAMOVANIE

2.1 Primárna úloha

Úlohou kvadratického programovania je nájsť minimum konvexnej kvadratickej funkcie viac premenných na množine prípustných riešení, ktorá je zadaná sústavou lineárnych rovníc a nerovníc. Takúto úlohu môžeme zapísať ako

$$\min_x \quad \frac{1}{2}x^T Cx + c^T x \quad (2.1)$$

$$\text{pri podmienkach} \quad Ax \leq b, \quad x \geq 0, \quad (2.2)$$

kde $x, c \in R^n$, $b \in R^m$ je tzv. vektor pravých strán, C je symetrická kladne semidefinitná matica stupňa n , A je (m, n) matica. Skrátene

$$\min_x \quad \left\{ \frac{1}{2}x^T Cx + c^T x \mid Ax \leq b, \quad x \geq 0 \right\}. \quad (\text{P})$$

Vektor x je prípustným riešením úlohy (P), ak spĺňa podmienky (2.2). Prípustné riešenie x , ktoré minimalizuje účelovú funkciu na množine prípustných riešení nazývame optimálne riešenie a budeme ho označovať \tilde{x} . Množinu prípustných riešení úlohy (P) označíme K ,

$$K = \{x \in R^n \mid Ax \leq b, \quad x \geq 0\}$$

a K_0 označíme množinu

$$K_0 = \{x \in R^n \mid Ax < b, x > 0\}.$$

Úloha (P) má nasledovné vlastnosti:

- (1) Každé lokálne riešenie úlohy (P) je zároveň jej optimálnym riešením[9].
- (2) Množina všetkých optimálnych riešení je konvexná[9].
- (3) Ak matica C je kladne definitná a množina prípustných riešení je neprázdna, potom (P) má jediné optimálne riešenie[9].

Každú nerovnosť vieme pridaním doplnkovej premennej zmeniť na rovnosť, pričom množina riešení zostane rovnaká. Nerovnostné ohraničenie z (P) môžeme zmeniť na rovnosť, pričom úloha (P) bude ekvivalentná úlohe

$$\min_{x,w} \left\{ \frac{1}{2}x^T Cx + c^T x \mid Ax + w = b, x \geq 0, w \geq 0 \right\}. \quad (\text{P1})$$

2.2 Podmienky optimality

Nutnými a postačujúcimi podmienkami, aby prípustné riešenie $\tilde{x} \in R^n$ bolo optimálnym riešením úlohy (P) sú Kuhn-Tuckerove podmienky optimality. Lagrangeova funkcia pre úlohu (P) má tvar

$$\Phi(x, u) = \frac{1}{2}x^T Cx + c^T x + u^T (Ax - b), \quad x \geq 0, u \geq 0. \quad (2.3)$$

Všimnime si, že úloha (P) zapísaná pomocou Lagrangeovej funkcie je

$$\min \left\{ \Phi(x, u) - \frac{\partial \Phi}{\partial u} u \mid \frac{\partial \Phi}{\partial u} \leq 0, x \geq 0 \right\}. \quad (2.4)$$

Kuhn-Tuckerove podmienky pre úlohu (P) sú

$$\begin{aligned} \frac{\partial \Phi}{\partial x} &\geq 0^T, & \frac{\partial \Phi}{\partial u} &\leq 0^T, \\ \frac{\partial \Phi}{\partial x} x &= 0, & \frac{\partial \Phi}{\partial u} u &= 0, \\ x &\geq 0, & u &\geq 0. \end{aligned} \tag{2.5}$$

To znamená, že

$$\begin{aligned} v &=: Cx + c + A^T u \geq 0, & -z &=: Ax - b \leq 0, \\ x^T(Cx + c + A^T u) &= 0, & u^T(Ax - b) &= 0, \\ x &\geq 0, & u &\geq 0, \end{aligned} \tag{2.6}$$

resp. zapísané pomocou nových premenných

$$-Cx - A^T u + v = c, \tag{2.7}$$

$$Ax + z = b, \tag{2.8}$$

$$x \geq 0, \quad u \geq 0, \quad v \geq 0, \quad z \geq 0, \tag{2.9}$$

$$x_i v_i = 0, \quad i = 1, \dots, n, \tag{2.10}$$

$$u_j z_j = 0, \quad j = 1, \dots, m. \tag{2.11}$$

Podmienka (2.8) je podmienkou primárnej prípustnosti, (2.7) je podmienkou duálnej prípustnosti, (2.10) a (2.11) sú tzv. podmienky komplementarity keďže zaručujú, aby vždy jedna z premenných bola nulová.

Veta 1: Kuhn-Tucker [2 str.121]. Bod $\tilde{x} \in R^n$ je optimálnym riešením úlohy (P) práve vtedy, ak existujú vektory $\tilde{u} \in R^m$, $\tilde{v} \in R^n$, $\tilde{z} \in R^m$ tak, že spolu s vektorom \tilde{x} vyhovujú podmienkam (2.7)-(2.11).

2.3 Duálna úloha

Duálna úloha k úlohe (2.4) zapísaná pomocou Lagrangeovej funkcie má tvar

$$\max \left\{ \Phi(x, u) - \frac{\partial \Phi}{\partial x} x \mid \frac{\partial \Phi}{\partial x} \geq 0, u \geq 0 \right\}. \quad (2.12)$$

Pre úlohu (P) je potom duálnou úlohou

$$\max \left\{ -\frac{1}{2}x^T Cx - b^T u \mid Cx + A^T u \geq -c, u \geq 0 \right\}. \quad (D)$$

Premenné x vystupujúce v úlohe (P) aj (D) sa nedajú stotožniť. Preto budeme používať v nasledujúcich vetách (y, u) ako neznáme v duálnej úlohe. O vzťahu medzi primárnou a duálnou úlohou hovoria ďalšie vety.

Veta 2: Slabá veta o dualite [2]. *Pre každé primárne prípustné riešenie x a každé duálne prípustné riešenie (y, u) platí*

$$\frac{1}{2}x^T Cx + c^T x \geq -\frac{1}{2}y^T Cy - b^T u.$$

Dôsledok [2]. *Ak pre každé primárne prípustné x a každé duálne prípustné riešenie (y, u) platí*

$$\frac{1}{2}x^T Cx + c^T x = -\frac{1}{2}y^T Cy - b^T u,$$

potom x a (y, u) sú optimálne riešenia (P), resp. (D).

Veta 3: Silná veta o dualite [2]. *Pre dvojicu vzájomne duálnych úloh kvadratického programovania platí:*

- a) *ak \tilde{x} je optimálne riešenie primárnej úlohy, potom existuje \tilde{u} také, že (\tilde{x}, \tilde{u}) je optimálne riešenie duálnej úlohy,*
- b) *ak (\tilde{x}, \tilde{u}) je optimálne riešenie duálnej úlohy a matica C je kladne definitná, potom zložka \tilde{x} je optimálne riešenie primárnej úlohy.*

V oboch úlohách platí, že v príslušných bodoch sa hodnoty účelových funkcií zhodujú. Podmienky optimality sú pre duálnu úlohu totožné s podmienkami optimality pre primárnu úlohu.

METÓDY VNÚTORNÉHO BODU

3.1. Centrálna trajektória

V tejto časti sa zameriame na algoritmus, ktorým dokážeme efektívne riešiť úlohu typu (1.15). Metódu, ktorú tu uvedieme sa nazýva *metóda sledujúca centrálnu trajektóriu*. Patrí do širokej triedy metód, ktorých hlavným spoločným znakom je postupné generovanie (v jednotlivých iteráciách) bodov (riešení pomocných úloh) z vnútra množiny prípustných riešení K_0 , ktoré konvergujú k riešeniu pôvodnej úlohy. Na základe toho získali pomenovanie metódy vnútorného bodu. Metóda sledujúca centrálnu trajektóriu je zrejme najjednoduchšia zo všetkých metód v triede metód vnútorného bodu. Najskôr sa sústredíme na definovanie "centrálnej trajektórie", ktorá sa vyskytuje v názve metódy. V tejto kapitole budeme používať neštandardné značenie. Malými písmenami (napr. x) budeme, ako zvyčajne, označovať vektory ($x = (x_1, \dots, x_n)^T$). Veľkými písmenami (napr. X) budeme označovať diagonálne matice, ktorých diagonálne prvky sú prvky príslušného vektora, napr.

$$X = \begin{pmatrix} x_1 & 0 & \dots & 0 \\ 0 & x_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & x_n \end{pmatrix}.$$

Takéto značenie sa bežne používa v problematike metód vnútorného bodu.

3.2. Bariérový problém

Hlavný problém, ktorý potrebujeme vyriešiť, sme sformulovali v predchádzajúcej kapitole a môžeme ho zapísať ako (P), teda

$$\min_x \{c^T x + \frac{1}{2}x^T C x \mid Ax \leq b, x \geq 0\}. \quad (3.1)$$

Vieme, že každú nerovnosť v ohraničeniach môžeme nahradiť rovnosťou pridaním nezápornej premennej, pričom sa úloha nezmení. Zmeňme nerovnostné ohraničenia v predchádzajúcej úlohe na rovnice (okrem premenných) pridaním nezápornej premennej w

$$\min_{x,w} \{c^T x + \frac{1}{2}x^T C x \mid Ax + w = b, x \geq 0, w \geq 0\}, \quad (3.2)$$

$x \in R^n$, $w \in R^m$ sú neznáme, $c \in R^n$, C symetrická kladne semidefinitná matica stupňa n , A matica ohraničení $m \times n$, $b \in R^m$ je vektor pravých strán. Ďalej sa môžeme pokúsiť odstrániť aj zvyšné nerovnosti v ohraničeniach (nezápornosť premenných). Každú nerovnosť nahradíme pridaním nového člena do účelovej funkcie nasledovne. Pre nezápornú premennú x_j pridáme do účelovej funkcie spojitú funkciu, ktorá dosahuje hodnotu mínus nekonečno ak x_j je záporná, dosahuje konečnú hodnotu ak x_j je kladná a pre x_j blížiac sa k nule dosahuje nekonečne veľké záporné hodnoty. Najjednoduchšou funkciou, ktorá spĺňa tieto požiadavky je logaritmická funkcia. Preformulujme teda úlohu (3.2) a pre každú nezápornú premennú doplníme do účelovej funkcie logaritmikus premennej násobený nezápornou konštantou μ .

$$\text{minimalizovať } (c^T x + \frac{1}{2} x^T C x - \mu \sum_{i=1}^n \ln x_i - \mu \sum_{j=1}^m \ln w_j) \quad (3.3)$$

pri podmienkach

$$Ax + w = b$$

x, w voľné.

Úloha (3.3) v skutočnosti nie je ekvivalentná s úlohou (3.2), ale je to v závislosti od parametra μ celá trieda úloh. Ak μ je blízke nule, potom sa (3.3) "blíži" k (3.2) v zmysle, že riešenia (3.3) sa "blížia" k riešeniu úlohy (3.2). Množina prípustných riešení je polyéder a známou vlastnosťou tohto polyédra je, že v ľubovoľnom bode steny je aspoň jedna premenná nulová. Preto účelová funkcia v (3.3) dosahuje na hranici množiny prípustných riešení nekonečne veľkú hodnotu a tak vytvára "bariéru". Problém (3.3) sa nazýva bariérový problém a účelová funkcia logaritmická bariérová funkcia. Bod vo vnútri množiny prípustných riešení má všetky premenné kladné, teda účelová funkcia má konečnú hodnotu. Minimum preto dosahuje vo vnútri množiny prípustných riešení. Toto riešenie je samozrejme pre každú hodnotu parametra μ iné. Ak sa na riešenia úlohy (3.3) pozrieme ako na funkciu parametra μ , množina optimálnych riešení vytvára trajektóriu cez vnútro množiny prípustných riešení. Túto trajektóriu nazývame *centrálna trajektória*.

3.3. Podmienky optimality v bariérovom probléme

Úloha (3.3) je optimalizačný problém s lineárnymi ohraňeniami typu rovností s voľnými premennými. Podmienky optimality pre takýto problém jednoducho

odvodíme z Lagrangeovej funkcie položením prvých parciálnych derivácií podľa všetkých premenných rovných nule. Lagrangeova funkcia je

$$\Phi(x, w, u) = c^T x + \frac{1}{2} x^T C x - \mu \sum_{i=1}^n \ln x_i - \mu \sum_{j=1}^m \ln w_j + u^T (Ax + w - b),$$

kde $x > 0$, $w > 0$. Z nej odvodené podmienky optimality sú

$$\begin{aligned} \frac{\partial \Phi}{\partial x_i} &= c_i + \sum_{j=1}^n \sigma_{ij} x_j - \mu \frac{1}{x_i} + \sum_{j=1}^m u_j a_{ji} = 0, & i = 1, \dots, n, \\ \frac{\partial \Phi}{\partial w_j} &= -\mu \frac{1}{w_j} + u_j = 0, & j = 1, \dots, m, \\ \frac{\partial \Phi}{\partial u_j} &= \sum_{i=1}^n a_{ij} x_i + w_j - b_j = 0, & j = 1, \dots, m. \end{aligned} \quad (3.4)$$

Rovnice (3.4) zapíšeme v maticovej forme:

$$\begin{aligned} -A^T u - Cx + \mu X^{-1} e &= c, \\ u &= \mu W^{-1} e, \\ Ax + w &= b, \end{aligned} \quad (3.5)$$

kde $x > 0$, $w > 0$. V (3.5) sme použili značenie, kde X je diagonálna matica, ktorej diagonálne prvky sú prvkami vektora x . Podobne W . Vektor $e = (1, \dots, 1)^T$ je jednotkový vektor. Zaveďme substitúciu $z = \mu X^{-1} e$:

$$\begin{aligned} z - Cx - A^T u &= c, & x > 0, \\ Ax + w &= b, & w > 0, \\ XZe &= \mu e, & z > 0, \\ UWe &= \mu e, & u > 0. \end{aligned} \quad (3.6)$$

Systém rovníc (3.6) je podobný systému (2.7)-(2.11). Prvá rovnica je podmienkou duálnej prípustnosti, druhá rovnica je podmienkou primárnej prípustnosti. Posledné dve rovnice sa podobajú na (2.10)-(2.11). Nula na pravej strane týchto rovníc je nahradená výrazom μe a nazývajú sa podmienky μ -komplementarity. Podmienky (3.6) dávajú systém $2m+2n$ rovníc s $2m+2n$ neznámymi (x, w, u, z) . Vyriešením tohto systému by sme získali optimálne riešenie (x_μ, w_μ) úlohy (3.3). Systém (3.6) je však nelineárny a preto ťažko riešiteľný. Množina bodov (riešení systémov (3.6)) $\{(x_\mu, w_\mu, u_\mu, z_\mu) : \mu > 0\}$ sa nazýva primárno-duálna centrálna trajektória a zohráva dôležitú úlohu v metódach vnútorného bodu. Existujú algoritmy, ktoré generujú postupnosť bodov na tejto trajektórii (alebo blízko nej) a táto postupnosť konverguje k optimálnemu riešeniu pôvodnej úlohy.

3.4. Nepřípustná primárno-duálna metóda sledujúca centrálnu trajektóriu

V tejto kapitole podrobne popíšeme algoritmus metódy, ktorá generuje body v okolí primárno-duálnej centrálnej trajektórie. Algoritmus môže začať z bodu, ktorý nie je ani primárne prípustným, ani duálne prípustným bodom. Vďaka tomu metóda získala pomenovanie nepřípustná primárno-duálna metóda sledujúca centrálnu trajektóriu. V numerických experimentoch bol použitý softvér, v ktorom je implementovaná spomenutá metóda vnútorného bodu. Algoritmus začína z ľubovoľného bodu $(x, w, u, z) > 0$ a iteratívne postupuje bodmi nasledovne:

- (1) Odhadne vhodnú hodnotu parametra μ .
- (2) Vypočíta smer $(\Delta x, \Delta w, \Delta u, \Delta z)$, ktorým sa posunieme "bližšie" k bodu na centrálnej trajektórii $(x_\mu, w_\mu, u_\mu, z_\mu)$.
- (3) Vypočíta parameter redukcie dĺžky kroku θ tak, že nový bod

$$\begin{aligned}\tilde{x} &= x + \theta \Delta x & \tilde{u} &= u + \theta \Delta u \\ \tilde{w} &= w + \theta \Delta w & \tilde{z} &= z + \theta \Delta z\end{aligned}$$

má aj naďalej kladné komponenty.

- (4) Nahradí aktuálny bod (x, w, u, z) novým bodom $(\tilde{x}, \tilde{w}, \tilde{u}, \tilde{z})$.

Toto bol zjednodušene popísaný princíp algoritmu. V ďalších častiach podrobne popíšeme jednotlivé kroky algoritmu, ktoré zhrnieme do prehľadnej schémy v poslednej časti tejto kapitoly.

3.4.1. Výpočet smerov

Naším cieľom je nájsť smer $(\Delta x, \Delta w, \Delta u, \Delta z)$ taký, že nový bod $(x + \Delta x, w + \Delta w, u + \Delta u, z + \Delta z)$, kde (x, w, u, z) je bod, v ktorom sa aktuálne nachádzame, je v istom zmysle lepší ako bod (x, w, u, z) . Ideálne by bolo, keby nový bod $(x + \Delta x, w + \Delta w, u + \Delta u, z + \Delta z)$ splňal rovnice

$$\begin{aligned} (z + \Delta z) - C(x + \Delta x) - A^T(u + \Delta u) &= c \\ A(x + \Delta x) + (w + \Delta w) &= b \\ (X + \Delta X)(Z + \Delta Z)e &= \mu e \\ (U + \Delta U)(W + \Delta W)e &= \mu e. \end{aligned} \quad (3.7)$$

Bod (x, w, u, z) poznáme, môžeme ho považovať za dáta a $(\Delta x, \Delta w, \Delta u, \Delta z)$ ako vektor neznámych. Prepíšeme systém (3.7) tak, že neznáme dáme na ľavú stranu a ostatné členy (vrátane nelineárnych neznámych) na pravú.

$$\begin{aligned} \Delta z - C\Delta x - A^T\Delta u &= c - z + Cx + A^T u &=: \tau \\ A\Delta x + \Delta w &= b - Ax - w &=: \rho \\ Z\Delta x + X\Delta z &= \mu e - XZe - \Delta X\Delta Ze &=: \gamma_x \\ W\Delta u + U\Delta w &= \mu e - UWe - \Delta U\Delta We &=: \gamma_u. \end{aligned} \quad (3.8)$$

Zaviedli sme nové označenie pravých strán $\tau, \rho, \gamma_x, \gamma_u$. Systém (3.8) je opäť systém nelineárnych rovníc, tentokrát pre "delta" premenné. Preto nebudeme riešiť nelineárny systém, ale iba určitú jeho lineárnu obmenu, ktorá nám umožní nájsť dobrý smer $(\Delta x, \Delta w, \Delta u, \Delta z)$. Táto obmena používa tzv. prediktor-korektor prístup. V kroku prediktor spravíme z (3.8) lineárny systém jednoduchým vynechaním všetkých nelineárnych členov z pravej strany. Pre parameter μ zvolíme hodnotu nula ($\mu = 0$). Novovytvorený systém sa potom rieši pre "delta" premenné.

$$\begin{aligned} \Delta z - C\Delta x - A^T\Delta u &= \tau \\ A\Delta x + \Delta w &= \rho \\ Z\Delta x + X\Delta z &= -XZe \\ W\Delta u + U\Delta w &= -UWe. \end{aligned}$$

Pomocou vypočítaných smerov sa určí parameter μ (určíme neskôr). "Delta" členy, ktoré sme v kroku prediktor vynechali z pravej strany systému, sa ako nové odhadnuté hodnoty $(\Delta x, \Delta w, \Delta u, \Delta z)$ dosadia naspäť do pravej strany systému

(3.8). Systém sa znovu rieši pre "delta" premenné na ľavej strane. Tento druhý krok sa nazýva korektor krok.

Hlavným výpočtovým problémom je riešenie systému (3.8) dvakrát v každej iterácii (jedenkrát krok prediktor, jedenkrát krok korektor). Treba si všimnúť, že systém (3.8) je veľký, riedky, indefinitný, lineárny systém. Ak preusporiadame rovnice

$$\begin{aligned}\Delta x + Z^{-1}X\Delta z &= \gamma_x \\ \Delta z - C\Delta x - A^T\Delta u &= \tau \\ A\Delta x + \Delta w &= \rho \\ \Delta u + W^{-1}U\Delta w &= \gamma_u,\end{aligned}$$

v maticovom zápise

$$\begin{bmatrix} -Z^{-1}X & -I & 0 & 0 \\ -I & C & A^T & 0 \\ 0 & A & 0 & I \\ 0 & 0 & I & W^{-1}U \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta x \\ \Delta u \\ \Delta w \end{bmatrix} = \begin{bmatrix} -\gamma_z \\ -\tau \\ \rho \\ \gamma_u \end{bmatrix}, \quad (3.9)$$

je to symetrický systém, čo ho predurčuje použiť postupnú elimináciu. Tento spôsob získava na dôležitosti pri komplikovanejšej úlohe (ďalšie ohraničenia, aj na premenné) Porovnanie takéhoto riešenia s inými prístupmi možno nájsť v [6].

Použitím pivotných prvkov $Z^{-1}X$ a $W^{-1}U$ eliminujeme členy Δz a Δw :

$$\begin{aligned}\Delta z &= X^{-1}Z(\gamma_z - \Delta x), \\ \Delta w &= U^{-1}W(\gamma_u - \Delta u).\end{aligned}$$

Substitúciou týchto výrazov získame systém:

$$\begin{bmatrix} -X^{-1}Z - C & -A^T \\ -A & U^{-1}W \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u \end{bmatrix} = \begin{bmatrix} \tau - X^{-1}Z\gamma_z \\ -\rho + U^{-1}W\gamma_u \end{bmatrix}. \quad (3.10)$$

Takýto systém sa v [7] nazýva redukovaný Karush-Kuhn-Tuckerov systém (KKT systém).

Systém (3.10) je opäť symetrický systém. Rozmer systému sme doteraz vedeli redukovať a zároveň sme dokázali zachovať symetriu. Nakoniec potrebujeme ešte doriešiť systém (3.10) pre neznáme Δx a Δu . Na to sa používa modifikovaný Choleského rozklad, ktorý bol prispôsobený na riešenie symetrických kvázidefinitných systémov. Príkladom takého systému je aj systém (3.10). Podrobnosti je možné nájsť v [6].

3.4.1.1. Newtonova metóda

Na prvý pohľad sa môže zdať riešenie systému (3.8) vynechaním nelineárnych členov z pravých strán v kroku prediktor neštandardné. V skutočnosti je to bežný postup pri riešení nelineárneho systému rovníc. Táto metóda sa nazýva *Newtonova metóda*.

Uvažujme zobrazenie $F : R^N \rightarrow R^N$, t.j.

$$F(\xi) = \begin{bmatrix} F_1(\xi) \\ F_2(\xi) \\ \vdots \\ F_N(\xi) \end{bmatrix}, \quad \xi = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_N \end{bmatrix}.$$

Úlohou je nájsť $\xi^* \in R^N$, koreň rovnice $F(\xi) = 0$. Newtonova metóda je iteračná metóda, ktorá rieši takýto problém. Jeden krok metódy môžeme popísať nasledovne:

Pre daný bod $\xi \in R^N$ je cieľom nájsť smer $\Delta\xi$ taký, že $F(\xi + \Delta\xi) = 0$. Pre nelineárnu funkciu F nie je ľahké nájsť taký vektor. Funkciu F aproximujeme prvými dvoma členmi Taylorovho rozvoja tejto funkcie:

$$F(\xi + \Delta\xi) \approx F(\xi) + F'(\xi)\Delta\xi, \quad (3.11)$$

kde

$$F'(\xi) = \begin{bmatrix} \frac{\partial F_1}{\partial \xi_1} & \frac{\partial F_1}{\partial \xi_2} & \cdots & \frac{\partial F_1}{\partial \xi_N} \\ \frac{\partial F_2}{\partial \xi_1} & \frac{\partial F_2}{\partial \xi_2} & \cdots & \frac{\partial F_2}{\partial \xi_N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_N}{\partial \xi_1} & \frac{\partial F_N}{\partial \xi_2} & \cdots & \frac{\partial F_N}{\partial \xi_N} \end{bmatrix}.$$

Aproximácia je lineárna v $\Delta\xi$. Položením pravej strany (3.11) rovnej nule dostaneme lineárny systém pre $\Delta\xi$

$$F'(\xi)\Delta\xi = -F(\xi).$$

Na konci každej iterácie aktualizujeme riešenie ξ novým riešením $\xi + \Delta\xi$. Proces pokračuje, kým nemáme "dobré" riešenie (t.j. $F(\xi) \approx 0$).

Teraz si všimnime systém (3.7). Riešením tohto systému je bod na centrálnej trajektórii. Položme

$$\xi = \begin{bmatrix} x \\ w \\ u \\ z \end{bmatrix} \quad a \quad F(\xi) = \begin{bmatrix} Ax + w - b \\ z - Cx - A^T u - c \\ XZe - \mu e \\ UWe - \mu e \end{bmatrix}.$$

Matica prvých derivácií tejto funkcie je:

$$F'(\xi) = \begin{bmatrix} A & I & 0 & 0 \\ -C & 0 & -A^T & I \\ Z & 0 & 0 & X \\ 0 & U & W & 0 \end{bmatrix}$$

a ak

$$\Delta\xi = \begin{bmatrix} \Delta x \\ \Delta w \\ \Delta u \\ \Delta z \end{bmatrix},$$

tak tento systém je

$$\begin{bmatrix} A & I & 0 & 0 \\ -C & 0 & -A^T & I \\ Z & 0 & 0 & X \\ 0 & U & W & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta w \\ \Delta u \\ \Delta z \end{bmatrix} = - \begin{bmatrix} Ax + w - b \\ z - Cx - A^T u - c \\ XZe - \mu e \\ UWe - \mu e \end{bmatrix}$$

a pre $\mu = 0$ je to presne systém, ktorý sa rieši v kroku prediktor.

3.4.2. Výpočet dĺžky kroku

Smery, ktoré sme v predchádzajúcej časti odvodili Newtonovou metódou, sme určili za predpokladu, že parameter redukcie dĺžky kroku $\theta > 0$ bol rovný jednej

$$\tilde{x} = x + \Delta x = x + \theta \Delta x.$$

Podobne aj pre $\tilde{w}, \tilde{u}, \tilde{z}$. Takýto spôsob posunutia do nového bodu však môže spôsobiť, že niektorý z vektorov $\tilde{w}, \tilde{u}, \tilde{z}$ dosiahne v niektorom komponente zápornú

hodnotu, čo je neprípustné. Preto je potrebné dĺžku kroku korigovať. Pre vektor $\tilde{x} = x + \theta \Delta x$ musíme garantovať, aby platilo:

$$x_i + \theta \Delta x_i > 0, \quad i = 1, \dots, n.$$

Presunutím Δx_i na druhú stranu nerovnosti a vydelením nerovnice θ a x_i (obidve sú kladné) zistíme, že musí platiť

$$\frac{1}{\theta} > -\frac{\Delta x_i}{x_i}, \quad i = 1, \dots, n.$$

Takéto nerovnosti musia spĺňať všetky komponenty vektorov $\tilde{x}, \tilde{w}, \tilde{u}, \tilde{z}$. Preto parameter θ musí spĺňať:

$$\frac{1}{\theta} = \max_{i,j} \left\{ -\frac{\Delta x_i}{x_i}, -\frac{\Delta w_j}{w_j}, -\frac{\Delta u_j}{u_j}, -\frac{\Delta z_i}{z_i} \right\}, \quad (3.12)$$

kde $\max_{i,j}$ znamená maximalizáciu cez všetky $i = 1, \dots, n, j = 1, \dots, m$. (3.12) ešte nezaručuje kladné hodnoty, preto požiadavku na parameter θ ešte upravíme:

$$\theta = \min \left\{ \frac{0,95}{\max_{i,j} \left\{ -\frac{\Delta x_i}{x_i}, -\frac{\Delta w_j}{w_j}, -\frac{\Delta u_j}{u_j}, -\frac{\Delta z_i}{z_i} \right\}}, 1 \right\}. \quad (3.13)$$

Pomocou takejto hodnoty parametra θ zaručíme splnenie požiadavky na vektory $\tilde{x}, \tilde{w}, \tilde{u}, \tilde{z}$. Teraz už môžeme aktualizovať riešenie:

$$x \longleftarrow x + \theta \Delta x$$

$$w \longleftarrow w + \theta \Delta w$$

$$u \longleftarrow u + \theta \Delta u$$

$$z \longleftarrow z + \theta \Delta z.$$

3.4.3. Odhad parametra μ

Vieme, že pre $\mu \rightarrow 0$ algoritmus generuje postupnosť bodov, ktoré konvergujú k optimálnemu riešeniu. V jednotlivých iteráciách potrebujeme znižovať hodnotu tohto parametra. Ak by sme zvolili pre μ príliš veľkú hodnotu, postupnosť bodov môže skonvergovať k analytickému stredu množiny prípustných riešení. Na druhej strane, ak μ bude príliš veľké, postupnosť sa môže vzdialiť príliš ďaleko od centrálnej trajektórie a skončí na hranici množiny prípustných riešení, ktorá však nie je optimálna. Preto potrebujeme vhodne zvoliť parameter μ , aby sme sa vyhli týmto extrémom.

Definujme pojem *duálna medzera* ako rozdiel hodnôt primárnej účelovej funkcie a duálnej účelovej funkcie na množinách prípustných riešení primárnej, resp. duálnej úlohy. Duálna medzera teda je:

$$\frac{1}{2}x^T Cx + c^T x - \left(-\frac{1}{2}x^T Cx - b^T u\right) = x^T Cx + c^T x + b^T u.$$

Duálna medzera je definovaná iba pre body primárne aj duálne prípustné. Takéto body spĺňajú ohraničenia

$$Ax + w = b,$$

$$-Cx - u^T A + z = c.$$

Ak prvú rovnicu vynásobíme vektorom u , druhú vektorom x a sčítame, dostaneme

$$u^T w + z^T x = x^T Cx + c^T x + b^T u.$$

Dôsledok slabej vety o dualite hovorí, že nulová duálna medzera implikuje optimálne riešenie. Našou snahou je duálnu medzeru znižovať. Ak v k -tej iterácii sme v bode (x_k, w_k, u_k, z_k) , tak na konci tejto iterácie sa posunieme do bodu $(x_{k+1}, w_{k+1}, u_{k+1}, z_{k+1})$. Pre tento bod by malo platiť

$$\begin{aligned}
& u_{k+1}^T w_{k+1} + z_{k+1}^T x_{k+1} = \\
& = (u_k + \theta \Delta u)^T (w_k + \theta \Delta w) + (z_k + \theta \Delta z)^T (x_k + \theta \Delta x) < u_k^T w_k + z_k^T x_k.
\end{aligned}$$

Ak upravíme tento výraz, získame odhad parametra μ

$$\begin{aligned}
& (u_k + \theta \Delta u)^T (w_k + \theta \Delta w) + (z_k + \theta \Delta z)^T (x_k + \theta \Delta x) = \\
& = u_k^T w_k + z_k^T x_k + \theta (w_k^T \Delta u + u_k^T \Delta w + z_k^T \Delta x + x_k^T \Delta z) + \theta^2 (\Delta u^T \Delta w + \Delta z^T \Delta x) < \\
& < u_k^T w_k + z_k^T x_k.
\end{aligned}$$

Prípustný bod (x_k, z_k, u_k, w_k) spĺňa rovnice (3.6). Platí, že $\Delta u^T \Delta w + \Delta z^T \Delta x = 0$. Preto

$$\theta (w_k^T \Delta u + u_k^T \Delta w + z_k^T \Delta x + x_k^T \Delta z) < 0.$$

Pretože $\theta > 0$ potom aj

$$w_k^T \Delta u + u_k^T \Delta w + z_k^T \Delta x + x_k^T \Delta z < 0.$$

Z (3.6) máme $x_k^T z_k = \mu n$ a $u_k^T w_k = \mu m$. Využijúc tieto vzťahy dostávame

$$\mu m - u_k^T w_k + \mu n - z_k^T x_k < 0.$$

Po úprave získame horný odhad parametra μ

$$\mu < \frac{u_k^T w_k + z_k^T x_k}{m + n}. \quad (3.14)$$

Pre bod (x_k, w_k, u_k, z_k) vieme ľahko vypočítať príslušnú hodnotu parametra μ

$$\mu = \frac{u_k^T w_k + z_k^T x_k}{m + n}. \quad (3.15)$$

Ak by sme pre získanie nového bodu použili takúto hodnotu parametra μ , nedošlo by k poklesu duálnej medzery. Naopak, ak zvolíme $\mu = 0$, dochádza k najväčšiemu poklesu duálnej medzery.

Aj napriek tomu, že v jednotlivých iteráciách nezískavame presné body z centrálnej trajektórie, používame (3.14) ako hodnotu pre parameter μ .

Systém (3.8) riešime v každej iterácii dvakrát (prediktor-korektor). Tieto dva kroky používajú dve rôzne hodnoty parametra μ . Krok prediktor sa snaží nájsť smer, v ktorom dochádza k najväčšiemu poklesu duálnej medzery, teda $\mu = 0$. Krok korektor koriguje smer, aby sme sa nedostali príliš ďaleko od centrálnej trajektórie. Keby sa v tomto kroku použila hodnota (3.14), nedošlo by k žiadnemu poklesu duálnej medzery. Preto sa parametru μ priraduje určitý násobok tejto hodnoty. Tento násobok sa v každej iterácii vypočítava dynamicky na základe parametra korekcie dĺžky kroku pre smery vypočítané v kroku prediktor. Teda parameter μ je

$$\mu = \left(\frac{\theta - 1}{\theta + 10} \right)^2 \frac{u_k^T w_k + z_k^T x_k}{m + n}. \quad (3.16)$$

3.4.4. Štart a koniec algoritmu

Na vstupe musíme zadať všetky potrebné dáta, t.j. c, C, A, b . Ďalej konštanty presnosti δ, ε . Na odštartovanie potrebujeme nastaviť počiatočné hodnoty premenných. Keďže algoritmus nepotrebuje prípustné riešenie, mohli by sme všetky voľné premenné nastaviť na nulu a všetky nezáporné premenné na jednotku. Lepšie je tieto hodnoty určiť trochu presnejšie, aby spĺňali niektoré rovnice. Neznáme x a u budú na začiatku riešením sústavy rovníc

$$\begin{bmatrix} -(C + I) & A^T \\ A & I \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} c \\ b \end{bmatrix}. \quad (3.17)$$

Ostatné premenné budú

$$\begin{aligned} w_i &= \max(|u_i|, 100), & i &= 1, \dots, m, \\ z_j &= \max(|x_j|, 100), & j &= 1, \dots, n, \end{aligned} \quad (3.18)$$

kde $|\cdot|$ je absolútna hodnota.

Pravidlá na ukončenie algoritmu sú:

Bod je primárne aj duálne prípustný a duálna medzera je menšia ako ε

Bod sa považuje za primárne prípustný ak platí:

$$\frac{\sqrt{\|\rho\|^2}}{\|b\| + 1} < \delta$$

Podobne, bod je duálne prípustný, ak spĺňa:

$$\frac{\sqrt{\|\sigma\|^2}}{\|c\| + 1} < \delta$$

Rovnako, ako optimálne riešenie, je dôležité sledovať neprípustnosť a neohraničenosť úlohy. Tieto javy sa dajú identifikovať na základe pozorovania primárnej a duálnej neprípustnosti. V normálnom stave neprípustnosť klesá monotónne. Ak primárna úloha je neprípustná, tak primárna neprípustnosť prestane klesať alebo dokonca rastie. Ak primárna úloha je neohraničená, tak duálna neprípustnosť prestane klesať alebo rastie. V takýchto prípadoch sa algoritmus zastaví.

3.4.5. Zhrnutie

Popísali sme si všetky jednotlivé kroky algoritmu. Teraz ho môžeme zhrnúť do prehľadnej schémy:

VSTUP: vektory c a b , matice C a A ,

konštanty presnosti $\delta = 10^{-6}$ a $\varepsilon = 10^{-8}$

inicializuj štartovací bod (x, w, u, z) podľa (3.18) a (3.17) $k=0$

- (1) $\mu = 0$
- (2) rieš (3.8), prediktor krok
- (3) vypočítaj μ podľa (3.15)
- (4) rieš (3.8), korektor krok
- (5) vypočítaj θ podľa (3.13)
- (6)

$\tilde{x}_{k+1} \leftarrow x_k + \theta \Delta x$	$\tilde{w}_{k+1} \leftarrow w_k + \theta \Delta w$
$\tilde{u}_{k+1} \leftarrow u_k + \theta \Delta u$	$\tilde{z}_{k+1} \leftarrow z_k + \theta \Delta z$
- (7) skontroluj podmienky ukončenia algoritmu, ak sú splnené (nájdené optimálne riešenie, neprípustnosť, neohraničenosť), koniec
- (8) inak $k = k + 1$, choď na (1)

4

NUMERICKÝ EXPERIMENT

4.1 LOQO

Cieľom diplomovej práce bolo overiť teoretické poznatky o metódach vnútorného bodu v praxi, zistiť dostupnosť softvéru na internete a otestovať ho na rôznych úlohách.

Voľne dostupný softvér, ktorý rieši úlohy kvadratického programovania nie je bežný, pravdepodobne aj preto, že je ho možné použiť aj na komerčné účely. Preto sa väčšinou licencie predávajú.

V tejto práci som sa rozhodol odskúšať softvér vyvinutý na americkej univerzite Princeton profesorom Vanderbeiom, ktorý je možné nájsť na adrese www.princeton.edu/~rvdb. Na komerčné použitie sa tento softvér tiež predáva, ale voľne prístupná je študentská verzia s určitými obmedzeniami na rozmer úlohy a na 60 dní je prístupná aj plná verzia bez limitov na dimenziu úlohy. Autor nazval spomenutý softvér LOQO a je napísaný pre operačný systém MS-DOS. LOQO rieši úlohu kvadratického programovania vo všeobecnom tvare

$$\min_x \quad f + c^T x + \frac{1}{2} x^T C x$$

pri podmienkach

$$b \leq Ax \leq b + r \tag{4.1}$$

$$l \leq x \leq u,$$

kde $x \in R^n$ je vektor neznámych, $c \in R^n$, $f \in R$, C je kladne semidefinitná a teda účelová funkcia je konvexná. A je $m \times n$ matica ohraničení, $b \in R^m$ sa nazýva vektor pravých strán, $r \in R^m$ je rozsah ohraničení, u a l sú horná, resp. dolná hranica premennej x . Predpokladá sa, že každý prvok b_i , $i = 1, \dots, m$ vektora b je konečné reálne číslo. Pre ostatné vektory platí

$$\begin{aligned} 0 &\leq r_i \leq \infty & i = 1, \dots, m, \\ -\infty &\leq l_j < \infty & j = 1, \dots, n, \\ -\infty &\leq u_j \leq \infty & j = 1, \dots, n. \end{aligned}$$

Algoritmus riešenia úlohy (4.1) sa dá odvodiť podobným spôsobom ako v kapitole 3. Podrobnosti možno nájsť v [6]. Úloha (3.1) je špeciálnym tvarom úlohy (4.1).

LOQO je systém na riešenie hladkých optimalizačných problémov. Úloha môže byť lineárna alebo nelineárna, konvexná alebo nekonvexná. Jediné obmedzenie je, že funkcie definované v úlohe musia byť hladké. Ak úloha je konvexná, tak LOQO nájde globálne optimálne riešenie. V opačnom prípade nájde lokálne optimálne riešenie.

Úlohu, teda vstupné dáta, je možné do softvéru zadať dvomi spôsobmi. Vo všeobecnosti je možné použiť modelovací jazyk AMPL [8]. Ak úloha je úlohou lineárneho programovania alebo úlohou kvadratického programovania s lineárnymi ohraničeniami, je možné dáta zadať vo vstupnom MPS súbore.

Na odštartovanie algoritmu LOQO nepotrebuje zadať štartovací bod. Ten sa dynamicky vypočíta ako v časti 3.4.4 (podrobnejšie v [6]).

Koniec algoritmu je určený niekoľkými pravidlami.

nájdené optimálne riešenie: podľa časti 3.4.4

nájdené suboptimálne riešenie: ak v niektorej iterácii je riešenie primárne aj duálne prípustné a v nasledujúcej iterácii stupeň neprípustnosti vzrastie (primárna alebo duálna), LOQO vyhlási riešenie z predchádzajúcej iterácie za suboptimálne. Príčinou je numerická nestabilita.

prekročený iteračný limit: LOQO povoľuje maximálne 500 iterácií.

Prax ukázala, že riešenie je získané za 10 až 60 iterácií.

primárna neprípustnosť: podľa 3.4.4

duálna neprípustnosť: podľa 3.4.4

Počas behu algoritmu sa pomocou nastaviteľných parametrov dá analyzovať priebeh výpočtu. Niektoré dôležité parametre sú:

Timing: nastavením na hodnotu 1 sa po skončení výpočtu vypíše čas (v sekundách) riešenia úlohy, prednastavená hodnota je 0 (čas sa nevypisuje)

Verbose: nastavením na hodnotu 2 sa počas výpočtu vypisujú na obrazovku údaje o jednotlivých iteráciách (hodnota primárnej a duálnej účelovej funkcie, primárna a duálna neprípustnosť a pod.)

Sigfig: špecifikuje na koľko miest sa musí zhodovať primárna a duálna účelová funkcia, aby sa riešenie prehlásilo za optimálne (prednastavená hodnota je 8)

Convex: určuje, že úloha je konvexná a nemusí sa zaoberať špeciálnymi prípadmi nekonvexnosti

Všetky parametre sú podrobne popísané v [8].

Úloha (3.1), ktorou sme sa v praktických výpočtoch zaoberali je úloha kvadratického programovania s lineárnymi ohraničeniami, preto je jednoduchšie dáta do softvéru zadávať v súbore MPS. MPS formát je štandardný formát používaný na zadávanie vstupov do optimalizačných softvérov. Je to textový súbor, ktorý musí spĺňať určité pravidlá, ktoré popíšeme na jednoduchom príklade. Uvažujme nasledujúcu úlohu kvadratického programovania

$$\begin{aligned}
\min \quad & 3x_1 - 2x_2 + x_3 - 4x_4 + \frac{1}{2}(x_3 - 2x_4)^2 \\
& x_1 + x_2 - 4x_3 + 2x_4 \geq 4 \\
& -3x_1 + x_2 - 2x_3 \leq 6 \\
& \quad +x_2 \quad \quad -x_4 = -1 \\
& \quad x_1 + x_2 - x_3 = 0 \\
& x_1 \text{ volné, } -100 \leq x_2 \leq 100, \quad x_3, x_4 \geq 0
\end{aligned}$$

NAME sample

ROWS

G r1

L r2

E r3

E r4

N obj

COLUMNS

x1	r1	1.
x1	r2	-3.
x1	r4	1.
x1	obj	3.
x2	r1	1.
x2	r2	1.
x2	r3	1.
x2	r4	1.
x2	obj	-2.
x3	r1	-4.
x3	r2	-2.
x3	r4	-1.
x3	obj	1.

```

      x4          r1          2.
      x4          r3         -1.
      x4          obj         -4.
RHS
      rhs         r1          4.
      rhs         r2          6.
      rhs         r3         -1.
BOUNDS
FR          x1
LO          x2          -100.
UP          x2          100.
QUADS
      x3          x3          1.
      x3          x4         -2.
      x4          x4          4.
ENDATA

```

Všetky heslá písané veľkými písmenami musia byť písané veľkými písmenami. Sú to kľúčové slová MPS formátu. Ostatné môžu byť písané veľkými alebo malými písmenami a reprezentujú informácie o úlohe. Zarovnanie stĺpcov je dôležité.

NAME udáva názov úlohy. Po skončení algoritmu sa výstup uloží do súboru s menom úlohy a príponou out.

V časti **ROWS** sa priradujú mená všetkým riadkom v úlohe a určuje sa, či je ohraničenie typu rovnosti E, typu "menšie rovné" L, typu "väčšie rovné" G alebo ide o lineárnu časť účelovej funkcie N.

Časť **COLUMNS** obsahuje páry premenná - riadok a hodnotu koeficientu všetkých nenulových prvkov v matici ohraničení a lineárnej časti účelovej funkcie. Pre každú premennú sa musia nenulové prvky zadávať spolu.

V časti **RHS** sa uvádzajú všetky nenulové pravé strany ohraničení.

V časti **BOUNDS** sú ohraničenia na premenné. FR označuje voľnú premennú,

UP označuje horné ohraničenie, LO je dolné ohraničenie. Ak sa premenná nešpecifikuje, predpokladá sa, že je nezáporná.

Kvadratické členy z účelovej funkcie sa uvádzajú v časti **QUADS**. Štruktúra je podobná časti **COLUMNS**, ale namiesto párov riadok-premenná sa uvádzajú páry premenná-premenná. Zapisujú sa iba prvky nad diagonálou a diagonálne prvky. Ostatné sa doplnia automaticky.

4.2 Úloha s fixnými transakčnými nákladmi

Riešená úloha s fixnými transakčnými nákladmi má tvar

$$\min_{x, x^+, x^-} -(\mu^T x - p^T x^+ - q^T x^-) + \frac{1}{2} x^T C x$$

$$l^T x = 1$$

$$x - x^+ + x^- = \hat{x}$$

$$x \geq 0, x^+ \geq 0, x^- \geq 0$$

$$x_i \leq 0.9, \quad i = 1, \dots, n$$

kde $\mu \in R^n$ je vektor očakávaných výnosov akcií v portfóliu, $x \in R^n$ je portfólio, $x^+, x^- \in R^n$ sú vektory nakúpených, resp. predaných akcií, p, q sú vektory transakčných nákladov na kúpu, resp. predaj akcií. C je kovariančná matica rozmeru $n \times n$. Ohraničenia v úlohe znamenajú, že všetky finančné prostriedky vložíme do investície ($l^T x = 1$) a pri kúpe akcií nepoužijeme žiadne nové investície a všetky prostriedky získané predajom akcií použijeme na nákup ďalších akcií ($x - x^+ + x^- = \hat{x}$).

Časovú náročnosť riešenia takejto úlohy som porovnával s úlohou bez transakčných nákladov obsahujúcou rovnaký počet akcií. Teda úlohou tvaru

$$\begin{aligned} \min_x \quad & -\mu^T x + \frac{1}{2} x^T C x \\ & l^T x = 1 \\ & x \geq 0 \\ & x_i \leq 0.9, \quad i = 1, \dots, n \end{aligned}$$

kde používame rovnaké označenie ako v predchádzajúcej úlohe.

LOQO bolo testované na dátach rôznych rozmerov. Najmenšia úloha obsahovala 100 aktív, najväčšia úloha mala 500 aktív. Ostatné úlohy obsahovali postupne 110, 120, 130, 140, 150, 160, 170, 180, 190, 200, 300 a 400 aktív. Pre každý rozmer som mal k dispozícii desať úloh. Takže celkovo som počítal 140 úloh pre fixné transakčné náklady a tiež 140 úloh bez transakčných nákladov. Treba si však uvedomiť, že úlohy s transakčnými nákladmi obsahujú vyššie spomenutý počet aktív, ale matematicky sa problém formuluje ako úloha s trojnásobným počtom premenných. Cieľom bolo zistiť časovú náročnosť nájdenia optimálneho riešenia pomocou algoritmu, v ktorom je implementovaná metóda vnútorného bodu popísaná v kapitole 3. Výsledný čas (v sekundách) riešenia úlohy daného rozmeru sa počítal ako aritmetický priemer časov všetkých desiatich úloh tohto rozmeru. V tabuľke sú uvedené výsledné priemerné časy riešenia úloh s fixnými transakčnými nákladmi spolu s úlohami bez transakčných nákladov.

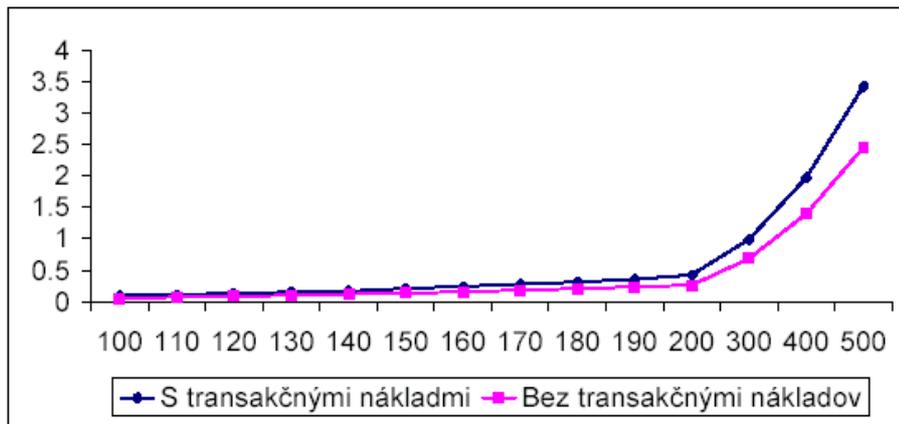
Tab. 1.

Počet aktív	Úloha	
	s transakčnými nákladmi	bez transakčných nákladov
100	0.0953	0.0564
110	0.1096	0.0723
120	0.1345	0.0847
130	0.1546	0.1016
140	0.1667	0.1186
150	0.2107	0.1391
160	0.2421	0.1576
170	0.2796	0.1829
180	0.3125	0.2097
190	0.3548	0.2374
200	0.4254	0.2625
300	0.9953	0.6888
400	1.9785	1.3987
500	3.4342	2.4597

Priemerný čas riešenia úlohy (v sekundách).

V prvom stĺpci je počet aktív v portfóliu, v druhom stĺpci je priemerný čas riešenia úlohy s transakčnými nákladmi a v treťom stĺpci je priemerný čas riešenia úlohy bez transakčných nákladov. Úloha s transakčnými nákladmi má trojnásobný počet premenných v porovnaní s úlohou bez transakčných nákladov. Zatiaľ čo úloha bez transakčných nákladov má iba jedno ohraničenie (okrem obmedzení na premenné), úloha s transakčnými nákladmi má $n + 1$ ohraňčení. Porovnaním časov riešení zodpovedajúcich úloh vidíme, že strojnásobením rozmeru úlohy čas vzrástol menej ako dvojnásobne. Dôvodom je, že matica v kvadratickom člene účelovej funkcie pre úlohu s transakčnými nákladmi vznikne z matice úlohy bez transakčných nákladov doplnením matice na potrebný rozmer nulami, čo umožňuje úlohu riešiť veľmi rýchlo. Porovnanie časov riešenia úloh s transakčnými nákladmi a bez transakčných nákladov je graficky znázornené na nasledujúcom grafe.

OBR. 7.



Grafické porovnanie priemerných časov riešení úloh s transakčnými nákladmi a úloh bez transakčných nákladov.

Záver

Táto diplomová práca sa zaoberá modernými metódami riešenia úloh kvadratického programovania. Problematika kvadratického programovania sa vo veľkej miere využíva vo finančnej sfére, pretože model optimálneho portfólia, ktorý je známy ako Markowitzov model optimálneho portfólia, je naformulovaný práve ako úloha kvadratického programovania.

V prvej kapitole sme odvodili Markowitzov model portfólia. Tento model sme rozšírili o transakčné náklady tak, že je ním možné modelovať reálnu situáciu na trhu. V tretej kapitole sme na úlohe optimálneho portfólia popísali algoritmus metódy vnútorného bodu. Tento algoritmus je implementovaný v softvéri, ktorý bol podrobne testovaný na mnohých úlohách s fixnými transakčnými nákladmi rôznych rozmerov. Profesor Best z kanadskej University of Waterloo poskytol na testovanie dáta úloh s variabilnými transakčnými nákladmi, ktoré sú dátami z reálnej situácie na trhu. Tieto úlohy viedli k riešeniu úlohy kvadratického programovania rozmerov až jedenásť násobne väčších ako je počet aktív v portfóliu. Bohužiaľ, tieto dáta vykazovali určité nezrovnalosti (neprípustnosť, resp. neohraničenosť úlohy) a pre nedostatok času sa nepodarilo analyzovať príčiny týchto nezrovnalostí. Práve riešenie takýchto úloh by bolo zaujímavé, pretože obsahujú niekoľkonásobne viac premenných. Pri riešení úloh tohto typu by sme získali modely s väčšími rozmermi ako modely s fixnými transakčnými nákladmi. V tomto smere teda vidíme možnosti pre ďalší výskum.

Hlavným prínosom tejto diplomovej práce je vysvetlenie algoritmu metódy, ktorá rieši úlohy kvadratického programovania v polynomiálnom čase a najmä otestovanie softvéru, ktorý je možné použiť pri riešení úloh kvadratického programovania, napr. aj pri hľadaní optimálneho akciového portfólia.

Literatúra

1. Z. Bodie, A. Kane, A.J. Marcus, *Investments*, The McGraw-Hill Companies, Inc., 1996.
2. M. Hamala, *Nelineárne programovanie*, ALFA - vydavateľstvo technickej a ekonomickej literatúry, 1976.
3. R. Tausinger, *Primárno-duálne metódy riešenia úloh kvadratického programovania*, Diplomová práca, MFF UK Bratislava, 2000.
4. T. Kapusta, *Optimalizácia portfólia*, Diplomová práca, MFF UK Bratislava, 2001.
5. M.J. Best, J.K. Kale, *Quadratic programming for large-scale portfolio optimization*, CRC Press, 1997.
6. R.J. Vanderbei, *LOQO: An interior point code for quadratic programming*, Princeton University, 1998.
7. R.J. Vanderbei, *Linear programming*, Kluwer, 1998.
8. R.J. Vanderbei, *LOQO user's manual*, <http://princeton.edu/rvdb/tex/loqo>, 1997.
9. J. Hloušková, *Structured portfolio optimization*, Dizertačná práca FMFI UK, Bratislava, 2001.
10. M.J Best, J. Hloušková, *An algorithm for portfolio optimization with transaction costs*.
11. M.J. Best, J Hloušková, *Portfolio selection and transaction costs*, 2000.
12. R.E. Bailey, *Economics of financial markets*, 2002.